MT 2500

AT Command Reference

GSM2418AT001

Version:1.03

04 June, 2012

enfora®
A Novatel Wireless Company

Terms Of Use

TERMS OF USE OF NEW MATERIALS - PLEASE READ CAREFULLY

From time to time, Enfora, in its sole discretion, may make available for download on its website (www.enfora.com), or may transmit via mail or email, updates or upgrades to, or new releases of, the firmware, software or documentation for its products (collectively, 'New Materials'). Use of such New Materials is subject to the terms and conditions set forth below, and may be subject to additional terms and conditions as set forth in Enfora's Technical Support Policy (posted on its website) and/or any written agreement between the user and Enfora.

All New Materials are provided AS IS. Enfora makes no warranty or representation with respect to the merchantability, suitability, functionality, accuracy or completeness of any such New Materials. The user of such New Materials assumes all risk (known or unknown) of such use. Enfora reserves all rights in such New Materials. The user shall have only a revocable and limited license to use such New Materials in connection with the products for which they are intended. Distribution or modification of any New Materials without Enfora's consent is strictly prohibited.

IN NO EVENT WILL ENFORA BE RESPONSIBLE FOR ANY INCIDENTAL, INDIRECT, CONSEQUENTIAL OR SPECIAL DAMAGES AS A RESULT OF THE USE OF ANY NEW MATERIALS. ENFORA'S MAXIMUM LIABILITY FOR ANY CLAIM BASED ON THE NEW MATERIALS SHALL NOT EXCEED FIFTY U.S. DOLLARS ($50).

Copyright

© 2012 Enfora, Inc. All rights reserved. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording or otherwise), or for any purpose, without the express written permission of Enfora, Inc.

Enfora and the Enfora logo are either registered trademarks or trademarks of Enfora, Inc. in the United States.

251 Renner Pkwy
Richardson, TX 75080 USA
972-633-4400
Phone: (972) 633-4400
Fax: (972) 633-4444
Email: info@enfora.com
www.enfora.com

Table of Contents

Introduction **1**

Document Scope 1

Platform Reference and Use 1

Command Syntax 1

Standard AT Commands 3

General Commands **4**

AT$DEVTYP Query the Device Type 4

AT$PKG Request Firmware Package 5

AT+CGMI Request Manufacturer Identification 7

AT+CGMM Request Manufacturer Model Identification 8

AT+CGMR Request Revision Identification 9

AT+CGSN Request IMEI 10

AT+GMI TA Manufacturer ID 11

AT+GMM TA Model ID 12

AT+GMR TA Revision Number 13

AT+GSN TA Serial Number 14

AT$SRN Module Serial Number 15

AT+WS46 Select Wireless Network 16

AT$MSPVER Query Current MSP Software Version 17

ATI Manufacturer Information About TA 18

ME TA Commands **19**

AT&C DCD Usage 19

AT&D DTR Usage 20

AT&F Set All TA Parameters to Factory Defined Configuration 21

AT&V Display Current Profile 22

AT&W Save Current Settings 23

AT$PWRMSG Power On Message 24
AT$URTWTO UART Wakeup Timeout 26
AT+CACM Accumulated Call Meter 27
AT+CAMM Accumulated Call Meter Maximum 28
AT+CEER Extended Error Reporting 29
AT+CFUN Set Phone Functionality 31
AT+CMEE Report Mobile Equipment Errors 32
AT+CPIN Enter PIN 33
AT+CPUC Price Per Unit and Currency Table 36
AT+CRES Restore Settings 37
AT+CSAS Save Settings 38
AT+CSCS Select TE Character Set 39
AT+CSTA Select Type of Address 40
AT+FCLASS GSM Class of Service 41
AT+GCAP Request Overall Capabilities for TA 42
AT+ICF TE-TA Character Framing 43
AT+IFC TE-TA Local Flow Control 44
AT+ILRR TE-TA Local Rate Reporting 45
AT+IPR Fixed TE-TA Data Rate 46
ATE Command Echo Mode 47
ATQ Result Code Suppression 48
ATS3 Command Line Termination Character 49
ATS4 Response Formatting Character 50
ATS5 Editing Character 51
ATV Response Format 52
ATX CONNECT Result 53
ATZ Set All TA Parameters to Default Configuration 54
PhoneBookCommands **55**

AT+CIND Indicator Control 55
AT+CLAE Set Language Event 57
AT+CLAN Set Language 58
AT+CPBF Find Phonebook Entries 60
AT+CPBR Read Phonebook Entries 61
AT+CPBS Select Phonebook Memory Storage 62
AT+CPBW Write Phonebook Entries 64
Call Control Commands **65**
+++ Escape Sequence 65
A - Manual Acceptance of a Network Request for PDP Context Activation 66
AT%CACM Query Accumulated Call Meter 67
AT%CPI Call Progress Information 68
AT%CAOC Query Current Meter Using PUCT 71
AT+CHLD Call Hold and Multiparty 72
AT+CHUP Hangup call 73
AT+CLCC List current calls 74
AT+CMUT Mute Control 76
AT+CPAS Phone Activity Status 77
AT+CR Service Reporting Control 78
AT+CRC Cellular Result Codes 79
AT+CSVM Set Voicemail Number 81
AT+STTONE GSM/GPRS Radio Modem-Start or Stop Generating a Tone 82
AT+VTS DTMF and Tone Generation 84
ATA Answer a Call 86
ATD Dial command 87
ATD> Originate Call Using Phonebook Memory 89
ATH Hook Control 91
ATP Pulse Dialing 92

ATO Return to Data State 93
ATS0 Automatic Response to a Network Request for PDP Context Activation 94
ATS0 Rings Before Automatic Answer 95
ATS10 Hang Up Delay 96
ATS6 Pause Before Blind Dialing 97
ATS7 Wait for Completion 98
ATS8 Dial Pause 99
ATT Tone Dialing 100
H Manual Rejection of a Network Request for PDP Context Activation 101
Network Service Commands 102
AT%ALS Alternating Line Service 102
AT+CAOC Advice of Charge 103
AT+CBST Select Bearer Service Type 104
AT+CCFC Call Forwarding Number and Condition 106
AT+CCUG Closed User Group 109
AT+CCWA Call Waiting 110
AT+CCWE Call Meter Maximum Event 112
AT+CIMI Request IMSI 113
AT+CLCK Facility Lock 114
AT+CLIP Calling Line Identification Presentation 117
AT+CLIR Calling Line Identification 119
AT+CMER Mobile Termination Event Reporting 121
AT+CMOD Call mode 124
AT+CNUM Subscriber Number 125
AT+COLP Connected Line Identification 126
AT+COPN Read Operator Names 127
AT+COPS Operator Selection 128
AT+CPOL Preferred Operator List 131

AT+CRLP Radio Link Protocol Parameters 132
AT+CSNS Single Numbering Scheme 133
Supplementary Service Commands **135**
AT+CSSN Supplementary Service Notifications 135
AT+CUSD Unstructured Supplementary Service 138
SMS Commands **140**
AT+CMGC Send Command 140
AT+CMGD Delete Message 141
AT+CMGF SMS Format 142
AT+CMGL List Messages 143
AT+CMGR Read Message 145
AT+CMGS Send Message 146
AT+CMGW Write Message to Memory 147
AT+CMSS Send Message from Storage 148
AT+CNMI New Message Indication to TE 149
AT+CPMS Preferred Message Storage 152
AT+CSCA Service Center Address 153
AT+CSCB Select Cell Broadcast Message Types 154
AT+CSDH Show Text Mode Parameters 155
AT+CSMP Set Text Mode Parameters 156
AT+CSMS Select Message Service 157
GPRS Commands **158**
AT+CGREG GPRS Network Registration Status 158
AT+CGACT PDP Context Activate or Deactivate 160
AT+CGANS Manual Response to a Network Request for PDP Context Activation 161
AT+CGATT GPRS Attach or Detach 162
AT+CGAUTO Automatic Response to a Network Request for PDP Context Activation 163
AT+CGCLASS GPRS Mobile Station Class 164

AT+CGDATA Enter Data State 165
AT+CGDCONT Define PDP Context 166
AT+CGEREP GPRS Event Reporting 168
AT+CGEV Deactivation Indication Reporting 170
AT+CGPADDR Show PDP Address 171
AT+CGQMIN Quality of Service Profile (Minimum Acceptable) 172
AT+CGQREQ Quality of Service Profile (Requested) 173
AT+CGSMS Select Service for MO SMS Messages 174
ATD Request GPRS Service 175
Audio Commands **176**
Basic Audio Commands 176
AT$VEAR EAR Gain 176
AT$VGR Radio Modem-Microphone Receiver Gain 177
AT$VGT Speaker Transmit Gain 178
AT$VLVL Speaker Volume 179
AT$VST Sidetone Volume 180
Advanced Audio Commands 181
AT$DFIR Configure Downlink FIR Coefficients 181
AT$MICAEC Echo Cancellation 183
AT$MICANR Ambient Noise Reduction Control 185
AT$MICBIAS Mic Bias 187
AT$MICES Echo Suppression Control 188
AT$MICESC Echo Suppressor Configuration 190
AT$PREAMP Set Uplink Voice Parameters 193
AT$SPKCFG Set Downlink Voice Parameters 195
AT$UFIR Configure Uplink FIR Coefficients 198
AT$VSELECT Voice Select 199
GPIO Commands **200**

AT$IGNDBNC Ignition Debounce 200
AT$IOCFG GPIO Configuration 201
AT$IOADC# Read Analog to Digital Converter 203
AT$IODBNC Debounce specified GPIO for the specified amount of time 204
AT$IOGPA GPIO Byte Control 205
AT$IOGP(x) GPIO Bit Control 207
AT$IOPULUP GPIO Pull-up Settings 208
AT$IOTAR Input/Output Transition Across Reset 209
AT$RPTADC Report ADC2 or GPIO9 210
IP Router Commands **211**
AT$GATEWAY Gateway IP 211
AT$DISC Disconnect Network Connection 211
AT$HOSTIF Configure Host to Modem Interface 213
AT$NETIP Display Network Assigned IP & DNS 215
AT$FLFILT - Filter ICMP Echo Request 215
AT$PING - Send ICMP Echo Request 216
Message Commands **218**
AT$APIOPT Enable API Optional Header Fields 218
AT$APIPWD API Password 221
AT$ATPASSWD Set authorization for AT commands for serial, SMS and API 222
AT$CHKIN Configure Device Check-in 223
AT$FRIEND Modem Friends 224
AT$MDMID Modem ID 227
AT$MLQSIZ Message Log Queue Size 228
AT$MSGLOGCL Message Log Clear 231
AT$MSGLOGDMP Dump Unsent Messages to Serial Port 232
AT$MSGLOGEN Message Log Enable 234
AT$MSGLOGRD Message Log Read Data 235

AT$MSGSND Message Send 238
AT$SMSDA Destination Address for SMS Messages 240
AT$SMSDAEN Enable/Disable AT Commands over SMS 242
AT$SNDMSG Send Test message 243
AT$UDPMSG Send and Receive UDP Messages 244
AT$WAKEUP Modem to Server Wakeup/Keep Alive 247
AT%SNCNT Query (or Reset) the Byte Counters (GPRS Only) 249
API Commands **250**
TCP API Commands 250
AT$TCPAPI TCP API Control 250
AT$TCPIDLETO TCP API Idle Timeout 252
AT$TCPRESTRT TCP API Restart 253
AT$TCPRETRYTO TCP API Retry Timeout 254
AT$TCPSRC TCP API Source Ports 255
AT$TCPSTATS TCP API Statistics 256
UDP API Commands 257
AT$ACKTM Acknowledgment Message Period & Retry Number 257
AT$UDPAPI Modem API Address 259
FOTA Commands **260**
AT$FOTAUPG Upgrade Current Firmware 260
AT$FOTACFG Configure Firmware Upgrade 261
AT$FOTAGET Get Firmware Upgrade OTA 263
AT$UPSTATUS Status of Last Upgrade 264
FTP Commands **265**
AT$ATEXEC Execute AT Command File 265
AT$FFS Generic Flash File System Access 267
AT$FTPABORT Aborts Current Data Operation with FTP Server 273
AT$FTPBLK Change Current FTP BLock Size 274

AT$FTPCFG Configure FTP parameters 275
AT$FTPCHDIR Change current working directory 276
AT$FTPCLOSE Closes the Connection to the FTP Server 277
AT$FTPDIR Directory Listing 278
AT$FTPGET Retrieve a Remote File 279
AT$FTPOPEN Opens FTP Connection 280
AT$FTPPUT Store a Local File On the FTP Server 281
AT$FTPR Outputs block of file data onto serial port 282
AT$FTPRST Restart the Last FTP GET Operation 283
Real-Time Clock Commands **284**
AT$RTCALRM Real Time Clock Alarm 284
AT$RTCCLRA Real Time Clock Clear Alarm 287
AT$RTCRSET RTC Report Reset State 288
AT$RTCTIME Real Time Clock Time 290
NITZ Commands **292**
AT$RTCUPD Update RTC with NITZ 292
AT%CNIV Generate URC with Network Name 293
AT%CTZV Generate URC with Date and Time 294
AT+CCLK Enable Setting and reading of RTC 295
AT+CSQ Signal Quality and Bit Error Rate 296
AT+CTZR Generate URC with Time Zone 299
AT+CTZU Enable Saving of Time Zone 300
Network Commands **301**
AT$AREG Auto Registration 301
AT$CGEER Get PDP Context Activation Reject Cause 304
AT$HBRST Automatic Modem Reset 306
AT$LOCI Location Information 308
AT$LUPREJ Get LUP Reject Cause 309

AT$MSCLS Set GPRS Multislot Class 311
AT$NETLST Network List 312
AT$NETMON Monitor Network Availability 314
AT$RESET Reset Modem 316
AT$TCPERR TCP Error Codes 316
AT%BAND Frequency Band Information 319
AT%CGAATT Automatic Attach and Detach Mode 321
AT%CGREG GPRS Extended Registration State 322
AT%NRG Network Registration and Service 324
AT+CREG Network Registration Info 326
AT$PDPBOT PDP Activation Backoff Time 328
Event Commands **329**
AT$ETSAV# Event Timer Save Configuration 329
AT$EVCID User Defined Incoming Call Number Event 330
AT$EVDEL Delete Event 332
AT$EVDELA Delete Event (All) 333
AT$EVDELR Delete a Range of Event Groups 334
AT$EVENT User Defined Input/Output 335
AT$EVGQRY Event Group Query 338
AT$EVNTRY Event Query 339
AT$EVTEST Generate Test Input Event 340
AT$EVTIM# User Defined Input Event Timers 341
AT$EVTIMQRY Event Counter 342
AT$EVTOFF Event Engine Disable 343
AT$EVTQRY Query the State or Value of the Specified Input Event 344
AT$STOATEV Store AT Command Events 345
AT$USRVAL User Value 347
SIM Toolkit Commands **348**

AT%SATC SET SIM Application Toolkit Configuration 348
AT%SATE Send SAT Envelope Command 350
AT%SATR Send SAT Command Response 351
AT%SATT Terminate SAT Command or Session 352
Miscellaneous Commands **353**
AT$AUDPLY Audio File Play 353
AT$AUDSTP Audio File Stop Playing 354
AT$ICCID Integrated Circuit Card ID 355
AT%CGPCO Set Type of Authentication, Username and Password 356
AT%CGPPP Negotiation Selection 358
AT%CSTAT Unsolicited SIM status 359
AT%EM Engineering Mode 360
AT%MEPD MEPD Configuration Data 361
AT%SIMIND SIM Status Indication 363
AT$SIMCFG SIM Configuration 365
AT%SLEEP Select level of sleep mode 366
AT+CPWD Change Password 368
AT$MDSTAT Query Modem Status 370
Accelerometer Commands **376**
Accelerometer 376
AT$ACCAC Accelerometer Auto-Calibration 377
AT$ACCAM Accelerometer 'Any Motion' Commands 378
AT$ACCCFG Accelerometer Configuration 380
AT$ACCEL Query Accelerometer Reading 382
AT$ACCFLT Accelerometer Filter Parameters 383
AT$ACCMGC Set Query Accelerometer orientation auto-calibration 385
AT$ACCORN Device Orientation 388
AT$ACCSAM Set/Query Accelerometer Sampling 389

AT$AUTOCFG Accelerometer Auto-Calibration Configuration 390
AT$MGCCFG Accelerometer MagiCal Configuration 392
GPS Commands **394**
AT$ALTOSI Set and Query GPS Overspeed Interval 394
AT$EXCACC Excessive Acceleration 395
AT$EXCDEC Excessive Deceleration 396
AT$GEOFNC Geo Fencing a Circular Area 397
AT$GFDBNC Set Geofence Debounce Count 399
AT$GFDEL Delete a Range of Geo-Fences 400
AT$GFIDX# Geo Fencing Index 401
AT$GPSCMD GPS Command 402
AT$GPSFD Restore GPS Filter Defaults 403
AT$GPSFLT GPS Speed Filter 404
AT$GPSLCL - Configure Sending of GPS Message to the Serial Port 411
AT$GPSLNA Enable/Disable Internal LNA 412
AT$GPSOSI Set and Query GPS Overspeed Interval 414
AT$GPSQUAL GPS Quality Filters 416
AT$GPSRD Read Current GPS ASCII data 418
AT$GPSRST Set and Query GPS Reset Interval 420
AT$GPSSAV Save GPS Data 422
AT$GPSVAL GPS Speed Validation 422
AT$TODOM Virtual Trip Odometer 423
AT$ODOMETER GPS Trip Odometer 425
AT$PLYCLR Clear Polygonal Geofence ? 426
AT$PLYFN# Geo Fencing a Polygonal Area ? 427
AT$PWRSAV Enable Power Save Mode 430
Event Tables **431**
Event Engine 431

Event Category 431
Event Type 432
Input Event Table 435
Output Event Table 444
User Variable Index Table 451
GPIO Flash Table 462
Bit-Field Tables 463
Bit-Field Table Selection 463
Bit-Field Table 0 – Legacy (0,0) 463
Bit 25 Binary Format Table 466
Bit 25 ASCII Format Table 467
Bit-Field Table 1 – (0,1) 469
Bit-Field Table 2 – (1,0) 473
Bit-Field Table 3 – (1,1) 479
Appendix A – Result Codes 481
Result Codes 481
Unsolicited Result Codes 481
SMS Unsolicited Result Codes 482
SAT Application Toolkit Result Codes 483
Appendix B – Error Codes 484
General Error Codes 484
GPRS Error Codes 486
SMS Error Codes (+CMS) 487
Release Causes for Extended Error Reporting (+CEER) 489
Extended Error Codes 491
Location Area Update Cause Codes ($LUPREJ)Routing Area Update Cause Codes ($RAUREJ) 492
Index 494

Introduction

Document Scope

The following documentation pertains to the AT Command Set to be used in conjunction with the Enfora Device identified in the title of this document.

Platform Reference and Use

The device will be referred to using various terms, to include: MS (Mobile Station), TA (Terminal Adapter), DCE (Data Communication Equipment), or ME (Mobile Equipment).

The device can be controlled via the use of a DTE (Data Terminal Equipment) platform by issuing the AT commands via a serial interface.

Command Syntax

The attention or "AT" prefix is required prior to entering any command. All commands require a carriage return or <CR> following the entry of the desired command. All command responses are encapsulated by a carriage return and line feed or <CR><LF>. The ASCII display of these characters is suppressed with only the modem response being presented.

In addition to terminating AT commands, the carriage return <CR> is also used to abort commands that are executing.

Most AT commands complete immediately so there is no opportunity to abort them, for instance ATI. However, Some commands like AT+COPS or AT+CFUN can actually take several seconds to complete. The AT command interface is said to be in execution mode when a command is running and has not returned a result code (OK/ERROR). A second <CR> entered while the AT command interface is in execution mode will abort the command and return the interface to command mode.

Some AT commands require additional input, for instance AT+CMGS. After terminating the AT+CMGS command with a <CR> the AT command interface enters line edit mode. While in line edit mode all characters are accepted except CNTL-Z. CNTL-Z terminates line edit mode and the AT command interface enters execution mode. Like before, at this point another <CR> will abort the command.

AT message concatenation can be done using the ; <semicolon> between commands.

The following examples demonstrate the potential usage of AT commands presented:

Type	Example	Description
Syntax Query	AT+GXXX=?	When entered will return the command format and value ranges.
Read Syntax	AT+GXXX?	When entered will return the current value assigned to the command.
Write Syntax	AT+GXXX=<value>,<value>,	When entered will set the command to specified value(s).
Execute Syntax	AT+GXXX	When entered will execute the specified command.
Command Concatenation	AT+CRC=1;S0=1	When entered it will execute both the CRC and S0 command.

Standard AT Commands

The following is the format in which all commands will be presented.

ATx(Command)	**Xxxxx (Command Description)**
Command Function	(Description of the command function)
Syntax Query	ATx=?
Syntax	ATx: (parameter1 name 1 – 15), (parameter2 name 1-10),…
Write Syntax	ATx=<value>,<value>[,<optional value>],…
Write Response	OK or ERROR
Read Syntax	ATx?
Read Response	<value>,<value>,…
Execute Syntax	ATx
Execute Response	OK, ERROR, or <value>
Unsolicited Response	
Parameter Values	
<Value 1>	ATx: (1-15),(1-10)
<Value 2>	
Notes	(Additional command notes)
Examples	

Note: Where applicable, the <value> responses provided for the READ and EXECUTE formats are modem default values. All efforts will be made by Enfora, Inc. to keep these values current in the documentation but will not be responsible for any differences that may occur as a result subsequent software builds and version enhancements.

General Commands

AT$DEVTYP Query the Device Type

AT$DEVTYP	Query the Device Type
Command Function	The $DEVTYP queries returns the product ID and software version.
Syntax Query	AT$DEVTYP=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$DEVTYP
Execute Response	$DEVTYP: <pid>,<swver> OK
Unsolicited Response	N/A
Parameter Values	
<pid>	Product ID
<swver>	4-digit hex Software Version value (for example, 2.1.4.0 would be displayed as 2140)
Notes	N/A
Example	N/A

AT$PKG Request Firmware Package

AT$PKG	Request Firmware Package
Command Function	This command is used to obtain the firmware package version.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$PKG
Execute Response	<baseline release>-<release offset> OK
Unsolicited Response	N/A
Parameter Values	
<baseline release>	1 to 99 - number associated with official production release

AT$PKG	Request Firmware Package
<release offset>	A = Controlled Availability : The 'A' release is a prerelease of the next production release and focuses on new features. These releases are rarely placed in manufacturing for shipment and the expectation is that customers move to the production release when available. C = Candidate (internal only) : This label is used to differentiate releases during the system test phase. Internal Release (IR) is by definition, "C1". It's possible that a 'C' release will be sent to a customer for the same reasons as the Controlled Availability release. This release type will not be available for order from manufacturing. D = Maintenance : A 'D' release includes the previous production release plus very specific bug fixes (no new features). Maintenance releases can be, and usually are, released to production. Not all products will necessarily have a maintenance release. For instance, if a Nitro has a GPS issue that requires a maintenance release, the OEM modules will not have a release. E = Engineering (internal only) : Although these are internal releases, they can be sent to specific customers to verify either a bug fix or new feature. These releases cannot be used by customers as a production release. This release type will not be available for order from manufacturing.
Notes	Return value is manufacturer specific.
Examples	AT$PKG? 48 This is a baseline production release AT$PKG? 48-D1 This is the baseline production release with additional bug fixes

AT+CGMI Request Manufacturer Identification

AT+CGMI	Request Manufacturer Identification
Command Function	This command is used to obtain the manufacturer identification information.
Syntax Query	AT+CGMI=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CGMI
Execute Response	Enfora, Inc. OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Return value is manufacturer specific.

AT+CGMM Request Manufacturer Model Identification

AT+CGMM	Request Manufacturer Model Identification
Command Function	This command is used to obtain the manufacturer model identification information.
Syntax Query	AT+CGMM=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CGMM
Execute Response **Response (if HS 1000)**	Enabler IIIG Modem OK DATA CARD OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Return value is manufacturer specific.

AT+CGMR Request Revision Identification

AT+CGMR	Request Revision Identification
Command Function	This command is used to obtain the manufacturer embedded firmware revision information.
Syntax Query	AT+CGMR=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CGMR
Execute Response	<revision> OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Return value is manufacturer specific.

AT+CGSN Request IMEI

AT+CGSN	Request IMEI
Command Function	This command is used to obtain the manufacturer International Mobile Equipment Identity (IMEI).
Syntax Query	AT+CGSN=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CGSN
Execute Response	0044008824900101 OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Return value is manufacturer specific. The TA returns the International Mobile station Equipment Identifier (IMEI).

AT+GMI TA Manufacturer ID

AT+GMI	TA Manufacturer ID
Command Function	TA returns information about the manufacturer.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+GMI
Execute Response **Response (if HS 1000)**	Enfora, Inc. OK 144. OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	N/A

AT+GMM TA Model ID

AT+GMM	TA Model ID
Command Function	TA returns manufacturer model identification.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+GMM
Execute Response	Enabler IIIG Modem OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	N/A

AT+GMR TA Revision Number

AT+GMR	TA Revision Number
Command Function	Returns software revision information.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+GMR
Execute Response	<revision> OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	N/A

AT+GSN TA Serial Number

AT+GSN	TA Serial Number
Command Function	This command is used to obtain the manufacturer International Mobile Equipment Identity (IMEI).
Syntax Query	AT+GSN=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+GSN
Execute Response	0044008824900101 OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Return value is manufacturer specific. The TA returns the International Mobile Equipment Identity (IMEI).

AT$SRN Module Serial Number

AT$SRN	Module Serial Number
Command Function	This command will return the serial number of the module.
Syntax Query	AT$SRN=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$SRN
Execute Response	$SRN: xxxxxxxxxxxxx
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Returned values are unique for each module

AT+WS46 Select Wireless Network

AT+WS46	Select Wireless Network
Command Function	This command is used to select the wireless network to operate with the TA.
Syntax Query	AT+WS46=?
Syntax	+WS46: (12) OK
Write Syntax	AT+WS46=<n>
Write Response	OK
Read Syntax	AT+WS46?
Read Response	+WS46: 12 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<n>	12 (GSM Digital Cellular)
Notes	Will provide available network interface support selection.

AT$MSPVER Query Current MSP Software Version

AT$MSPVER	Query Current MSP Software Version
Command Function	This command allows the user to query the current software ID of the MSP430.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	AT$MSPVER?
Read Response	$MSPVER: version OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<version>	0 -5 = version ID unknown (MSP430 has not communicated the version to the modem) 6 – 255 = version ID received from the MSP430
Notes:	N/A

ATI Manufacturer Information About TA

ATI	Manufacturer Information About TA
Command Function	List manufacturer.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	ATI
Execute Response **Response (if HS 1000)**	Enfora, Inc. OK 144. OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	N/A

ME TA Commands

AT&C DCD Usage

AT&C	DCD Usage
Command Function	Controls the Data Carrier Detect signal.
Syntax Query	N/A
Syntax	N/A
Write Syntax	AT&C<value>
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<value>	0 = DCD always on 1 = DCD matches the state of the remote modem's data carrier
Notes	Some Enfora products have no DCD hardware line.

AT&D DTR Usage

AT&D	DTR Usage
Command Function	This command controls the Data Terminal Ready signal.
Syntax Query	N/A
Syntax	N/A
Write Syntax	AT&D<value>
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<value>	0 = Ignore DTR 1 = Modem switches from DATA to COMMAND mode when DTR switches to off 2 = When DTR switches to off, disconnect the call. Automatic answer CSD call is disabled while DTR remains off"
Notes	Some Enfora products have no DTR hardware line.

AT&F Set All TA Parameters to Factory Defined Configuration

AT&F	Set All TA Parameters to Factory Defined Configuration
Command Function	Set All TA Parameters to Factory Defined Configuration
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT&F
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	

AT&V Display Current Profile

AT&V	Display Current Profile
Command Function	This command allows the user to view the settings in the current profile.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT&V
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	This command may report serial port baud rates. However, the baud rate cannot be changed and is not used on this device and is reported by this command for backward compatibility only.

AT&W Save Current Settings

AT&W	Save Current Settings
Command Function	This command allows the user to save the current settings in memory.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT&W
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	To ensure successful completion of the command, do not issue additional commands until 'OK' is returned.

	Warning: **Users should avoid sending AT&W immediately before a modem reset. A minimum of a few seconds should be allowed between issuing the command and a modem reset.**

AT$PWRMSG Power On Message

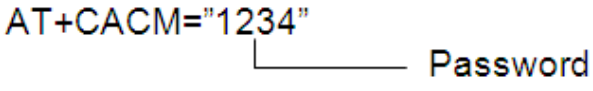
AT$PWRMSG	Power On Message
Command Function	This command allows the user to change the default Power up message.
Syntax Query	AT$PWRMSG=?
Syntax	$PWRMSG: "pwr up msg"
Write Syntax	AT$PWRMSG=”new pwr up message”
Write Response	OK
Read Syntax	AT$PWRMSG?
Read Response	$PWRMSG: "AT-Command Interpreter ready"
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<message>	New Power up Message
<0,1>	Optional parameter that will remove the message altogether. Please see the examples at the end of this command.
Notes	AT$PWRMSG=”” will restore the power up message to the factory default. The power up message can be a maximum of 40 characters.

AT$PWRMSG	Power On Message
Example	AT$PWRMSG? $PWRMSG: "AT-Command Interpreter ready" AT$RESET AT-Command Interpreter ready AT$PWRMSG ="Ready To Go" AT$PWRMSG? $PWRMSG: "Ready To Go" AT$RESET Ready To Go AT$PWRMSG ="" AT$PWRMSG? $PWRMSG: "AT-Command Interpreter ready" The second parameter is optional. AT$PWRMSG="New Powerup Msg" sets 'New Powerup Msg' as powerup msg AT$PWRMSG="New Powerup Msg",0 same as previous AT$PWRMSG=,1 removes the powerup msg AT$PWRMSG="some text",1 removes the powerup msg (ignores text) AT$PWRMSG="" sets powerup msg back to default string AT$PWRMSG="",0 same as previous

AT$URTWTO UART Wakeup Timeout

AT$URTWTO	UART Wakeup Timeout
Command Function	This sets the number of seconds the UART will prevent the modem from going to deep sleep after it receives a character.
Syntax Query	AT$URTWTO=?
Syntax	$URTWTO: (0-60) OK
Write Syntax	AT$URTWTO=<seconds>
Write Response	OK
Read Syntax	AT$URTWTO?
Read Response	$URTWTO: 10 OK
Execute Syntax Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<seconds>	0 – Sets the UART Wakeup time to the default value of 10 seconds. 1-60 – Sets the UART Wakeup time to the specified number of seconds.
Notes:	N/A

AT+CACM Accumulated Call Meter

AT+CACM	Accumulated Call Meter
Command Function	Set command resets the Advice of Charge related accumulated call meter value in SIM file EFACM. ACM contains the total number of home units for both the current and preceding calls. SIM PIN2 is usually required to reset the value.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	AT+CACM?
Read Response	+CACM: “000000” OK
Execute Syntax	AT+CACM=<passwd>
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	<passwd>: string type; SIM PIN2
Notes	Used in conjunction with AT+CAOC and AT+CAMM
Example:	

AT+CAMM Accumulated Call Meter Maximum

AT+CAMM	Accumulated Call Meter Maximum
Command Function	Set command sets the Advice of Charge related accumulated call meter maximum value in SIM file EFACMmax. ACMmax contains the maximum number of home units allowed to be consumed by the subscriber. When ACM (refer +CACM) reaches ACMmax calls are prohibited (see also GSM 02.24 [26]). SIM PIN2 is usually required to set the value.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CAMM=<acmmax>,<passwd>
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	
<acmmax>	string type; accumulated call meter maximum value similarly coded as <ccm> under +CAOC; value zero disables ACMmax feature
<passwd>	string type; SIM PIN2
Notes	Used in conjunction with AT+CACM and AT+CAOC.

AT+CEER Extended Error Reporting

AT+CEER	Extended Error Reporting
Command Function	This command is used to control the display of extended result codes for last unsuccessful call setup, in-call modification, last call release, last short message, or last GPRS session.
Syntax Query	AT+CEER=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CEER
Execute Response	+CEER: < DEFBY >,<ORIGSIDE>,<ORIGIN_ENTITY>,<VALUE>[,ERROR DESCRIPTION] OK
Unsolicited Response	N/A
Parameter Values	
<DEFBY>	(defined by) 0 = Standard 1 = Enfora
<ORIGSIDE>	(originating side) 0 = Network 1 = MS

AT+CEER	Extended Error Reporting
<ORIGIN_ENTITY>:	0 = SIM 1 = ACI 2 = RLP 3 = RR 4 = MM 5 = CC 6 = SS 7 = SMSCP 8 = SMSRP 9 =- SMSTP 10 = GMM 11 = SM 12 = FAD 13 = T30 14 = GRR 15 = PPP 16 = LLC 17 = SNDCP 18 = PKTIO 19 -= PSI
<VALUE>	See AT+CEER Table in Appendix B
<ERROR DESCRIPTION>	Optional extended error description
Notes	N/A

AT+CFUN Set Phone Functionality

AT+CFUN	Set Phone Functionality
Command Function	Set command selects the level of functionality <fun> in the ME. Level "full functionality" is where the highest level of power is drawn. "Minimum functionality" is where minimum power is drawn.
Syntax Query	AT+CFUN=?
Syntax **Response (If HS 1000)**	+CFUN: (0,1,4), (0) OK +CFUN: (0,1,2,3,4), (0,1) OK
Write Syntax	AT+CFUN=<fun>,<rst>
Write Response	OK
Read Syntax	AT+CFUN?
Read Response	+CFUN: 1 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<fun>	0 = minimum functionality 1 = full functionality 2 = disable phone transmit RF circuits only (HS 1000 Only) 3 = disable phone receive RF circuits only (HS 1000 Only) 4 = disable phone both transmit and receive RF circuits
<rst>	0 = Do not reset ME
Notes	Once the modem has left the minimum functionality state, it will respond to AT+CFUN? with +CFUN: 1 regardless of whether the modem has reached full functionality yet. (Not applicable to HS 1000)

AT+CMEE Report Mobile Equipment Errors

AT+CMEE	Report Mobile Equipment Errors
Command Function	Set command disables or enables the use of result code +CME ERROR: <err> as an indication of an error relating to the functionality of the ME. When enabled, ME related errors cause +CME ERROR: <err> final result codes to be returned, instead of the default ERROR final result code. ERROR is returned normally when error is related to syntax, invalid parameters, or TA functionality.
Syntax Query	AT+CMEE=?
Syntax	+CMEE: (0-2) OK
Write Syntax	AT+CMEE=<n>
Write Response	OK
Read Syntax	AT+CMEE?
Read Response	+CMEE: 0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<n>	0 = Disable +CME ERROR 1 = Enable +CME result code and username values 2 = Enable +CME result code and ME verbose values
Notes	

AT+CPIN Enter PIN

AT+CPIN	Enter PIN
Command Function	Set command sends to the ME a password that is necessary before it can be operated (SIM PIN, SIM PUK, PH-SIM PIN, etc.). If no PIN request is pending, no action is taken towards ME and an error message, +CME ERROR, is returned to TE. If the PIN required is SIM PUK or SIM PUK2, the second pin is required. This second pin, <newpin>, is issued to replace the old pin in the SIM.
Syntax Query	AT+CPIN=?
Syntax	OK
Write Syntax	AT+CPIN=<”pin”>,[<”newpin”>]
Write Response	
Read Syntax	AT+CPIN?
Read Response	+CPIN: <code> OK or +CME ERROR: <err>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	

AT+CPIN	Enter PIN
<code>	READY = ME is not pendingfor any password SIM PIN = ME is waiting SIM PIN to be given SIM PUK = ME is waiting SIM PUK to be given PH-SIM PIN = ME is waiting phone-to-SIM card password to be given PH-FSIM PIN = ME is waiting phone-to-very first SIM card password to be given PH-FSIM PUK = ME is waiting phone-to-very first SIM card unblocking password to be given SIM PIN2 = ME is waiting SIM PIN2 to be given (this <code> is recommended to be returned only when the last executed command resulted in PIN2 authentication failure (i.e. +CME ERROR: 17); if PIN2 is not entered right after the failure, it is recommended that ME does not block its operation) SIM PUK2 = ME is waiting SIM PUK2 to be given (this <code> is recommended to be returned only when the last executed command resulted in PUK2 authentication failure (i.e. +CME ERROR: 18); if PUK2 and new PIN2 are not entered right after the failure, it is recommended that ME does not block its operation) PH-NET PIN = ME is waiting network personalization password to be given PH-NET PUK = ME is waiting network personalization unblocking password to be given PH-NETSUB PIN = ME is waiting network subset personalization password to be given PH-NETSUB PUK = ME is waiting network subset personalization unblocking password to be given PH-SP PIN = ME is waiting service provider personalization password to be given PH-SP PUK = ME is waiting service provider personalization unblocking password to be given PH-CORP PIN = ME is waiting corporate personalization password to be given PH-CORP PUK = ME is waiting corporatepersonalization

AT+CPIN	Enter PIN
Note	Commands which interact with ME that are accepted when ME is pending SIM PIN,SIM PUK, or PH-SIM are: +CGMI, +CGMM, +CGMR, +CGSN, D112; (emergency call),+CPAS, +CFUN, +CPIN, After power on the modem needs 20-25 seconds to initialize and completely read the SIM. * If AT$AREG=1, and PIN is enabled, the modem will not complete the auto registration process until after the PIN has been entered (AT+CPIN). Quotation marks are optional when entering <pin> or <newpin>.

AT+CPUC Price Per Unit and Currency Table

AT+CPUC	Price Per Unit and Currency Table
Command Function	Set command sets the parameters of Advice of Charge related price per unit and currency table in SIM file EFPUCT.
Syntax Query	N/A
Syntax	N/A
Write Syntax	AT+CPUC=<currency>,<ppu>,<passwd>
Write Response	OK
Read Syntax	AT+CPUC?
Read Response	AT+CPUC: “ “, “ “ OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<currency>	string type; three-character currency code (e.g. "GBP", "DEM"); character set as specified by command Select TE Character Set +CSCS
<ppu>	string type; price per unit; dot is used asa decimal separator (e.g. "2.66")
<passwd>	string type; SIM PIN2
Notes	N/A

AT+CRES Restore Settings

AT+CRES	Restore Settings
Command Function	Restores message service settings from non-volatile memory to active memory.
Syntax Query	AT+CRES=?
Syntax	+CRES: (0-3) OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CRES
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Retrieves profiles stored using AT+CSAS.

AT+CSAS Save Settings

AT+CSAS	Save Settings
Command Function	Saves active message service commands into non-volatile memory.
Syntax Query	AT+CSAS=?
Syntax	+CSAS: (0-3) OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CSAS
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	AT+CRES retrieves stored profiles. Settings specified in commands Service Center Address +CSCA, Set Message Parameters +CSMP and Select Cell Broadcast Message Types +CSCB are saved.

AT+CSCS Select TE Character Set

AT+CSCS	Select TE Character Set
Command Function	This command is used to select the terminal equipment character set.
Syntax Query	AT+CSCS=?
Syntax	+CSCS: "GSM", "IRA" , "PCCP437" , "PCDN" , "8859-1" , "HEX" , “UCS2” OK
Write Syntax	AT+CSCS=<chset>
Write Response	OK
Read Syntax	AT+CSCS?
Read Response	+CSCS: “PCCP437” OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<chset>	"GSM" "IRA" "PCCP437" "PCDN" "8859-1" "HEX" "UCS2"
Notes	Values are based on character set support.

AT+CSTA Select Type of Address

AT+CSTA	Select Type of Address
Command Function	This command is used to select the type of number to be used for further dialing commands.
Syntax Query	AT+CSTA=?
Syntax	+CSTA: (129,145) OK
Write Syntax	AT+CSTA=<n>
Write Response	OK
Read Syntax	AT+CSTA?
Read Response	+CSTA: 129 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<n>	129 - Dialing string without International Access Code character “+”) 145 - Dialing string with International Access Code character “+”
Notes	N/A

AT+FCLASS GSM Class of Service

AT+FCLASS	GSM Class of Service
Command Function	This command puts the TA into a particular mode of operation (data, voice etc.). This causes the TA to process information in a manner suitable for that type of information (rather than for other types of information).
Syntax Query	AT+FCLASS=?
Syntax	0,2.0,8 OK
Write Syntax	AT+FCLASS=<mode>
Write Response	OK
Read Syntax	AT+FCLASS?
Read Response	0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = Data 2.0 = fax class 2 (ITU-T T.32 [12] and TIA-592) 8 = Voice
Notes	N/A

AT+GCAP Request Overall Capabilities for TA

AT+GCAP	Request Overall Capabilities for TA
Command Function	TA returns a list of additional capabilities
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+GCAP
Execute Response **Response (if HS 1000)**	+GCAP: +CIS707-A, +MS, +ES, +DS, +FCLASS OK +GCAP: +CGSM OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	N/A

AT+ICF TE-TA Character Framing

AT+ICF	TE-TA Character Framing
Command Function	This command determines the number of data/stop/parity bits that will be used by the TA serial interface.
Syntax Query	AT+ICF=?
Syntax	+ICF: (1-6), (0-3) OK
Write Syntax	AT+ICF=<format>,<parity>
Write Response	OK
Read Syntax	AT+ICF?
Read Response	+ICF: 3 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<format>	1 = 8 data, 2 stop, no parity 2 = 8 data, 1 stop,1 parity 3 = 8 data, 1 stop, no parity 4 = 7 data, 2 stop, no parity 5 = 7 data, 1 stop, 1 parity 6 = 7 data, 1 stop, no parity
<parity>	0 = odd 1 = even 2 = mark
Notes	If no parity is specified in <format>, then <parity> is ignored.

AT+IFC TE-TA Local Flow Control

AT+IFC	TE-TA Local Flow Control
Command Function	This command determines the TE/TA flow control interface.
Syntax Query	AT+IFC=?
Syntax	+IFC: (0,2), (0,2) OK
Write Syntax	AT+IFC=<DCE_by_DTE>,<DTE_by_DCE>
Write Response	OK
Read Syntax	AT+IFC?
Read Response	+IFC: 0,0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<DCE_by_DTE>	0 = None 1 = Xon/Xoff (not supported) 2 = Hardware Flow Control
<DTE_by_DCE>	0 = None 1 = Xon/Xoff (not supported) 2 = Hardware Flow Control
Notes	N/A

AT+ILRR TE-TA Local Rate Reporting

AT+ILRR	TE-TA Local Rate Reporting
Command Function	State Control
Syntax Query	AT+ILRR=?
Syntax	+ILRR: (0,1) OK
Write Syntax	AT+ILRR=<value>
Write Response	OK
Read Syntax	AT+ILRR?
Read Response	+ILRR: 0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<value>	0 = Disable reporting of local port rate 1 = Enable reporting of local port rate
Notes	N/A

AT+IPR Fixed TE-TA Data Rate

AT+IPR	Fixed TE-TA Data Rate
Command Function	Determines the data rate of the TA serial interface.
Syntax Query	AT+IPR=?
Syntax	+IPR: (0, 1200,2400,4800,9600,14400,19200,28800,38400,57600,115200), (300,600,230400,460800,921600) OK
Write Syntax	AT+IPR=<rate>
Write Response	OK
Read Syntax	AT+IPR?
Read Response	+IPR: 115200 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<rate>	0,300,600,1200,2400,4800,9600,14400,19200,28800,38400,57600,115200, 230400,460800,921600
Notes	When changing the value of AT+IPR, the new baud rate is effective immediately. In order to properly save the new setting and communicate with the modem, the user must change the baud rate of the communicating device to the new baud rate before any more communication with the modem can be accomplished. While under CMUX, AT+IPR will not change the modem's baud rate. For USB virtual COM ports, the +IPR query command will always return 0 (autobaud). For the +IPR Write Syntax, the rate will be accepted but has no effect on the virtual COM port.

ATE Command Echo Mode

ATE	Command Echo Mode
Command Function	Determines whether the TA echoes characters typed locally.
Syntax Query	N/A
Syntax	N/A
Write Syntax	ATE<value>
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<value>	0 =Do not echo characters locally 1= Echo characters locally
Notes	N/A

ATQ Result Code Suppression

ATQ	Result Code Suppression
Command Function	Determines whether or not the TA transmits any result code to the TE.
Syntax Query	N/A
Syntax	N/A
Write Syntax	ATQ<value>
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<value>	0 =DCE transmits result codes 1 =Result codes are suppressed and not transmitted
Notes	N/A

ATS3 Command Line Termination Character

ATS3	Command Line Termination Character
Command Function	Determines the character recognized by the TA to terminate an incoming command line.
Syntax Query	ATS3=?
Syntax	S3(0-127) OK
Write Syntax	ATS3=<n>
Write Response	OK
Read Syntax	ATS3?
Read Response	013 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	N/A
Notes	N/A

ATS4 Response Formatting Character

ATS4	Response Formatting Character
Command Function	Determines the character generated by the TA for result code and information text.
Syntax Query	ATS4=?
Syntax	S4(0-127) OK
Write Syntax	ATS4=<n>
Write Response	OK
Read Syntax	ATS4?
Read Response	010 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	N/A
Notes	N/A

ATS5 Editing Character

ATS5	Editing Character
Command Function	Determines the character recognized by the TA as a request to delete the preceding character form the command line.
Syntax Query	ATS5=?
Syntax	S5(0-127) OK
Write Syntax	ATS5=<n>
Write Response	OK
Read Syntax	ATS5?
Read Response	008 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	N/A
Notes	N/A

ATV Response Format

ATV	Response Format
Command Function	Determines the DCE response format, with or without header character, and the use of numerical results code.
Syntax Query	N/A
Syntax	N/A
Write Syntax	ATV<value>
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<value>	0 = DCE transmits limited headers and trailers and numeric result codes 1 = DCE transmits full headers and trailers and verbose response text
Notes	N/A

ATX CONNECT Result

ATX	CONNECT Result
Command Function	Determines whether or not the TA transmits particular result codes.
Syntax Query	N/A
Syntax	N/A
Write Syntax	ATX<value>
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<value>	0 = Short result code format 1 = Long result code format
Notes	For UDP and TCP PAD operation, setting of ATX1 will display the network assigned IP after the CONNECT or LISTEN message.

ATZ Set All TA Parameters to Default Configuration

ATZ	Set All TA Parameters to Default Configuration
Command Function	Set All TA Parameters to Default Configuration.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	ATZ
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Non volatile memory is not reset.

PhoneBookCommands

AT+CIND Indicator Control

AT+CIND	Indicator Control
Command Function	Set command is used to set the values of MT indicators. Read command returns the status of MT indicators. If MT is not currently reachable, +CME ERROR: <err> is returned. Refer subclause 9.2 for <err> values. Test command returns pairs, where string value <descr> is a maximum 16 character description of the indicator and compound value is the allowed values for the indicator. If MT is not currently reachable, +CME ERROR: <err> is returned. Refer subclause 9.2 for <err> values.
Syntax Query	AT+CIND=?
Syntax	+CIND: (“signal”,(0-5)),(“smsfull”,(0-1)) OK
Write Syntax	AT+CIND=<signal>,<smsfull>
Write Response	OK
Read Syntax	AT+CIND?
Read Response	+CIND: <signal>,<smsfull> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<signal>	signal quality (0-5)

AT+CIND	Indicator Control
<smsfull>	A short message memory storage in the MT has become full and a short message has been rejected (2), has become full(1), or memory locations are available (0).
Notes	N/A

AT+CLAE Set Language Event

AT+CLAE	Set Language Event
Command Function	This command is used to enable/disable unsolicited result code +CLAV: <code>.
Syntax Query	AT+CLAE=?
Syntax	+CLAE: (0,1) OK
Write Syntax	AT+CLAE=<mode>
Write Response	OK
Read Syntax	AT+CLAE?
Read Response	+CLAE: 0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = Disable 1 = Enable
Notes	N/A

AT+CLAN Set Language

AT+CLAN	Set Language
Command Function	This command sets the language in the ME. The set-command must confirm the selected language with the MMI-module in the ME. If setting fails, a ME error, +CME ERROR: <err> is returned. Refer to subclause 9.2 for <err> values.
Syntax Query	AT+CLAN=?
Syntax	+CLAN: en, fr, de, it, es, pt, no, el, pl, in, cs, zh, ar OK
Write Syntax	AT+CLAN=<code>
Write Response	OK
Read Syntax	AT+CLAN?
Read Response	+CLAN: en OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	

AT+CLAN	Set Language
<code>	“en” = English “fr” = French “de” = German “it” = Italian “es” = Spanish “pt” = Portuguese “no” = Norwegian “el” = Greek “pl” = Polish “in” = Indonesian “cs” = Czech “zh” = Chinese “ar” = Arabic
Notes	N/A

AT+CPBF Find Phonebook Entries

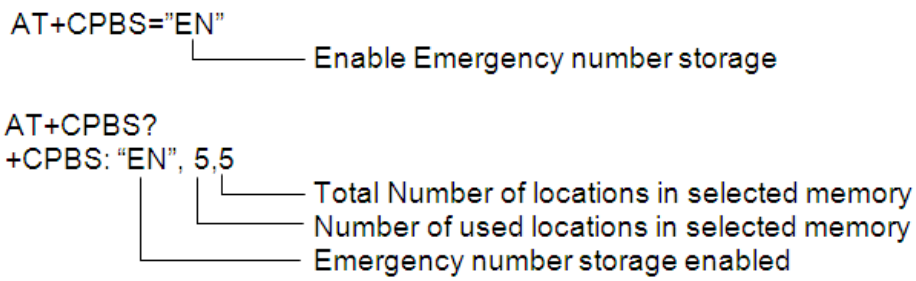
AT+CPBF	Find Phonebook Entries
Command Function	Execution command returns phonebook entries (from the current phonebook memory storage selected with +CPBS) which alphanumeric field start with string <findtext>.
Syntax Query	AT+CPBF=?
Syntax	+CPBF: <nlength>,<tlength> OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CPBF=<”findtext”>
Execute Response	+CPBF: <index1>,<number>,<type>,<text><CR><LF>+CBPF: <index2>,<number>,<type>,<text>… OK
Unsolicited Response	N/A
Parameter Values	
<index1>,<index2>	integer type values in the range of location numbers of phonebook memory
<number>	string type phone number of format <type>
<type>	type of address octet in integer format
<findtext>,<text>	string type field of maximum length <tlength>; character set as specified by command Select TE Character Set +CSCS
<nlength>	integer type value indicating the maximum length of field <number>
<tlength>	integer type value indicating the maximum length of field <text>
Notes	This command will find an entry within the storage facility that is set with AT+CPBS.
Example:	AT+CPBF=”office” +CPBF: 10,”19725551212”,129,”office”

AT+CPBR Read Phonebook Entries

AT+CPBR	Read Phonebook Entries
Command Function	Execution command returns phonebook entries in location number range <index1>... <index2> from the current phonebook memory storage selected with +CPBS. If <index2> is left out, only location <index1> is returned.
Syntax Query	AT+CPBR=?
Syntax	+CPBR: (1-250), 44,16 OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CPBR=<index1>,<index2>,…
Execute Response	+CPBR: <index1>,<number>,<type>,<text> OK
Unsolicited Response	N/A
Parameter Values	
<index1>, <index2>, <index>	integer type values in the range of location numbers of phonebook memory
<number>	string type phone number of format <type>
<type>	type of address octet in integer format
<text>	string type field of maximum length <tlength>; character set as specified by command Select TE Character Set +CSCS
<nlength>	integer type value indicating the maximum length of field <number>
<tlength>	integer type value indicating the maximum length of field <text>
Notes	This command will read the storage facility that is set with AT+CPBS.

AT+CPBS Select Phonebook Memory Storage

AT+CPBS	Select Phonebook Memory Storage
Command Function	Set command enables or disables sending of unsolicited result codes from TA to TE in the case of key pressings, display changes, and indicator state changes.
Syntax Query	AT+CPBS=?
Syntax	+CPBS: ("EN","BD","FD","DC","LD","RC","LR","MT","AD","SM","SD","MC", "LM","ON","UD") OK
Write Syntax	AT+CPBS=<storage>
Write Response	OK
Read Syntax	AT+CPBS?
Read Response	+CPBS: <storage>,<used>,<total> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	

AT+CPBS	Select Phonebook Memory Storage
<storage>	"EN" - SIM (or ME) emergency number "FD" - SIM fixed-dialing-phonebook "LD" - SIM last-dialing-phonebook "BD" - SIM barred-dialing phonebook "SD" - SIM service numbers “DC” - MT dialed calls list “RC” - MT received calls list "LR" - Last received numbers (nonstandard) “MT” - combined MT and SIM/UICC phonebook "AD" - Abbreviated dialing numbers (nonstandard) "LM" - Last missed numbers (nonstandard) “MC” - MT missed (unanswered received) calls list "SM" - comb. of fixed and abbrev. dialing phonebook (nonstandard) “ON” - active application in the UICC (GSM or USIM) or SIM card (or MT) own numbers (MSISDNs) list “UD” - User defined
<used>	integer type value indicating the number of used locations in selected memory
<total>	integer type value indicating the total number of locations in selected memory
Notes	To read the storage facilities, the correct storage must be written to first and then read.
Example	

AT+CPBW Write Phonebook Entries

AT+CPBW	Write Phonebook Entries
Command Function	Execution command writes phonebook entry in location number <index> in the current phonebook memory storage selected with +CPBS.
Syntax Query	AT+CPBW=?
Syntax	+CPBW: (1-250),44,(128-201), 16 OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	AT+CPBW?
Read Response	+CPBW: <index>, [<nlength>],<types>,[<tlength>] OK
Execute Syntax	AT+CPBW=<index>,<number>,<type>
Execute Response	<text> OK/+CME ERROR: <err>
Unsolicited Response	N/A
Parameter Values	
<index>	integer type values in the range of location numbers of phonebook memory
<number>	string type phone number of format <type>
<type>	type of address octet in integer format; default 145 when dialling string includes international access code character "+", otherwise 129
<text>	string type field of maximum length <tlength>; character set as specified by command Select TE Character Set +CSCS
<nlength>	integer type value indicating the maximum length of field <number>
<tlength>	integer type value indicating the maximum length of field <text>
Notes	This command will write to the storage facility that is set with AT+CPBS.
Example:	AT+CPBW=10,”16662326602”,129,”Toms Office”

Call Control Commands

+++ Escape Sequence

+++	Escape Sequence
Command Function	This command allows a user to escape out of data mode to command mode in a CSD call or from connect or listen mode to command mode in a GPRS call
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	+++
Execute Response	OK or no carrier
Unsolicited Response	NA
Parameter Values	N/A
Notes	The escape sequence requires a guard period of 1 second before and after entering +++. Otherwise the +++ will be considered data and forwarded as data. For CSD, to end the call ATH or AT+CHUP must be entered. To return to data mode issue ATO command.

A - Manual Acceptance of a Network Request for PDP Context Activation

A	Manual Acceptance of a Network Request for PDP Context Activation
Command Function	The V.25ter 'A' (Answer) command may be used to accept a network request for a PDP context activation announced by the unsolicited result code RING.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	A
Execute Response	Connect
Unsolicited Response	N/A
Parameter Values	N/A
Notes	N/A

AT%CACM Query Accumulated Call Meter

AT%CACM	Query Accumulated Call Meter Using PUCT
Command Function	Returns the current value of the accumulated call meter, calculated with the values given by the price per unit and currency table stored in SIM. Refer subclause 9.2 of [GSM 07.07] for possible <err> values.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT%CACM
Execute Response	%CACM: <cur>,<price> OK
Unsolicited Response	N/A
Parameter Values	
<cur>	string type; three-character currency code (e.g. "GBP", "DEM"); character set as specified by command Select
<price>	string type; calculated price value of accumulated call meter; dot is used as a decimal separator (e.g. 2.66)
Notes	N/A

AT%CPI Call Progress Information

AT%CPI	Call Progress Information
Command Function	This command refers to call progress information, which is indicated by the network during call establishment. The set command enable/disables the presentation of unsolicited notification result codes from TA to TE. When <mode>=1 and a call progress information is received during a call establishment, intermediate result code %CPI: <cId>,<msgType>,<ibt>,<tch> is sent to TE. <cId> identifies the call in the call table. The value of <msgType> describes the layer 3-message type that was used to transfer the call progress information. The state of TCH assignment and the use of in-band tones for that call can be monitored by the values of <ibt> and <tch>. Test command returns values supported by the TA as compound value.
Syntax Query	AT%CPI=?
Syntax	%CPI: (0-4) OK
Write Syntax	AT%CPI=<mode>
Write Response	OK
Read Syntax	AT%CPI?
Read Response	%CPI: 0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	

AT%CPI	Call Progress Information
<mode>	(parameter sets/shows the result code presentation status in the TA) 0 = disable 1 = enable 2 =status 3 = append cause and ALS bearer state to unsolicited result code 4 = appends an Advanced Cause Code (For Experienced Users Only)
<cId>	integer type; call identification number as described in GSM 02.30 subclause 4.5.5.1
<msgType>	(layer 3 message type) 0 = setup message 1 = disconnect message 2 = alert message 3 = call proceed message 4 = synchronization message 5 = progress description message 6 = connect 7 = reset request for call reestablishment 8 = reset confirm for call reestablishment 9 = call release 10 = call reject 11 = mobile originated call setup 12 = call hold
<ibt>	(status of the usage of in-band tones) 0 = no in-band tones 1 = in-band tones

AT%CPI	Call Progress Information
<tch>	(TCH assignment) 0 = TCH not assigned 1 = TCH assigned
Notes	%CPI=4 appends an Advanced Cause Code (For Experienced Users Only)

AT%CAOC Query Current Meter Using PUCT

AT%CAOC	Query Current Meter Using PUCT
Command Function	Returns the current value of the current call meter, calculated with the values given by the price per unit and currency table stored in SIM. Refer subclause 9.2 of [GSM 07.07] for possible <err> values.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT%CAOC
Execute Response	%CAOC: <cur>,<price> OK
Unsolicited Response	N/A
Parameter Values	
<cur>	string type; three-character currency code (e.g. "GBP", "DEM"); character set as specified by command Select
<price>	string type; calculated price value of accumulated call meter; dot is used as a decimal separator (e.g. 2.66)
Notes	N/A

AT+CHLD Call Hold and Multiparty

AT+CHLD	Call Hold and Multiparty
Command Function	This command controls the supplementary services Call Hold, MultiParty and Explicit Call Transfer. Calls can be put on hold, recovered, released, added to conversation and transferred.
Syntax Query	AT+CHLD=?
Syntax	+CHLD: (0, 1, 1x, 2, 2x, 3, 4) OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CHLD=<n>
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	
<n>	0 = Terminate all held calls; or set UDUB (User Determined User Busy) for a waiting call, i.e. reject the waiting call. 1 = Terminate all active calls (if any) and accept the other call (waiting call or held call) 1X = Terminate the active call X (X= 1-7) 2 = Place all active calls on hold (if any) and accept the other call (waiting call or held call) as the active call 2X = Place all active calls except call X (X= 1-7) on hold 3 = Add the held call to the active calls 4 = Connects the two calls and disconnects the subscriber from both calls (ECT).
Notes	Call Hold, MultiParty and Explicit Call Transfer are only applicable to teleservice 11(Speech Telephony).

AT+CHUP Hangup call

AT+CHUP	Hangup call
Command Function	This command is used to end all active calls.
Syntax Query	AT+CHUP=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CHUP
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Default value will be 0. AT&F, restore factory defaults will reset this value to 0.

AT+CLCC List current calls

AT+CLCC	List current calls
Command Function	Returns list of current calls of ME. If command succeeds but no calls are available, no information response is sent to TE.
Syntax Query	AT+CLCC=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CLCC
Execute Response	[+CLCC: <id1>,<dir>,<stat>, <mode>,<mpty> [,<number>, <type>[,<alpha>]] [<CR><LF>+CLCC: <id2>,<dir>,<stat>,<mode>, <mpty>[,<number>,<type>[,<alpha>]][...]]] OK
Unsolicited Response	N/A
Parameter Values	
<idx>	integer type; call identification number as described in GSM 02.30 [19] subclause 4.5.5.1; this number can be used in +CHLD command operations
<dir>	0 = mobile originated (MO) call 1 = mobile terminated (MT) call
<stat>	(state of the call): 0 = active 1 = held 2 = dialling (MO call) 3 = alerting (MO call) 4 = incoming (MT call) 5 = waiting (MT call)

AT+CLCC	List current calls
<mode>	(bearer/teleservice): 0 = voice 1 = data 2 = fax (fax not supported) 3 = voice followed by data, voice mode 4 = alternating voice/data, voice mode 5 = alternating voice/fax, voice mode 6 = voice followed by data, data mode 7 = alternating voice/data, data mode 8 = alternating voice/fax, fax mode 9 = unknown
<mpty>	0 = call is not one of multiparty (conference) call parties 1 = call is one of multiparty (conference) call parties
<number>	string type phone number in format specified by <type>
<type>	type of address octet in integer format (refer GSM 04.08 [8] subclause 10.5.4.7)
<alpha>	string type alphanumeric representation of <number> corresponding to the entry found in phonebook; used character set should be the one selected with command Select TE Character Set +CSCS
Notes	N/A

AT+CMUT Mute Control

AT+CMUT	Mute Control
Command Function	This command is used to enable and disable the uplink voice muting during a voice call.
Syntax Query	AT+CMUT=?
Syntax	+CMUT: (0,1) OK
Write Syntax	AT+CMUT=<value>
Write Response	OK
Read Syntax	AT+CMUT?
Read Response	+CMUT: 0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<value>	0 = mute off 1 = mute on
Notes	N/A

AT+CPAS Phone Activity Status

AT+CPAS	Phone Activity Status
Command Function	Execution command returns the activity status <pas> of the ME. It can be used to interrogate the ME before requesting action from the phone. Test command returns values supported by the ME as a compound value.
Syntax Query	AT+CPAS=?
Syntax	+CPAS: (0-5) or +CME ERROR: <err> OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CPAS
Execute Response	AT+CPAS: <pas> OK
Unsolicited Response	N/A
Parameter Values	
<pas>	0 = Ready (ME allows commands from TA/TE) 1 = Unavailable (ME does not allow commands from TA/TE) 2 = Unknown (ME is not guaranteed to respond to instructions) 3 = Ringing (ME is ready for commands from TA/TE, but the ringer is active) 4 = Call in progress (ME is ready for commands from TA/TE, but a call is in progress) 5 = Asleep (ME is unable to process commands from TA/TE because it is in a low functionality state)
Notes	N/A

AT+CR Service Reporting Control

AT+CR	Service Reporting Control
Command Function	This command is used to control the display of intermediate result code (+CR <serv>) status.
Syntax Query	AT+CR=?
Syntax	+CR: (0,1) OK
Write Syntax	AT+CR=<mode>
Write Response	OK
Read Syntax	AT+CR?
Read Response	+CR: 0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = disable 1 = enable
<serv>	ASYNC = asynchronous transparent SYNC = synchronous transparent REL ASYNC = asynchronous non-transparent REL SYNC = synchronous non-transparent
Notes	If enabled, the intermediate result code is transmitted at the point during connect negotiation at which the TA has determined which speed and quality of service will be used, before any error control or data compression reports are transmitted, and before any final result code (e.g. CONNECT) is transmitted.

AT+CRC Cellular Result Codes

AT+CRC	Cellular Result Codes
Command Function	This command is used to control the display of extended incoming call information.
Syntax Query	AT+CRC=?
Syntax	+CRC: (0,1) OK
Write Syntax	AT+CRC=<mode>
Write Response	OK
Read Syntax	AT+CRC?
Read Response	+CRC: 0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = disable 1 = enable

AT+CRC	Cellular Result Codes
<type>	ASYNC = asynchronous transparent SYNC = synchronous transparent REL ASYNC = asynchronous non-transparent REL SYNC = synchronous non-transparent FAX = facsimile (TS 62) VOICE = normal voice (TS 11) VOICE/ XXX = voice followed by data (BS 81) (XXX is ASYNC, SYNC, REL ASYNC or REL SYNC) ALT VOICE/ XXX = alternating voice/data, voice first (BS 61) ALT XXX/VOICE = alternating voice/data, data first (BS 61) ALT VOICE/FAX alternating voice/fax, voice first (TS 61) ALT FAX/VOICE alternating voice/fax, fax first (TS 61)
Notes	When enabled, an incoming call is indicated to the TE with unsolicited result code +CRING: <type> instead of the normal RING.

AT+CSVM Set Voicemail Number

AT+CSVM	Set Voicemail Number
Command Function	The number to the voice mail server is set with this command. The parameters <number> and <type> can be left out if the parameter <mode> is set to 0.
Syntax Query	AT+CSVM=?
Syntax	+CSVM: (0,1),(129,145,161) OK
Write Syntax	AT+CSVM=<mode>,<number>,<type>
Write Response	OK
Read Syntax	AT+CSVM?
Read Response	+CSVM: 0, " ",129 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = Disable the voice mail number 1 = Enable the voice mail number
<number>	string type;Character string <0..9,+>
<type>	integer type; Type of address octet 129 = ISDN / telephony numbering plan, national / international unknown 145 = ISDN / telephony numbering plan, international number 161 = ISDN / telephony numbering plan, national number
Notes	The voicemail number is set in the SIM by the service provider. Care should be taken when entering this command. If the voicemail number is lost or does not work, contact your service provider for the correct voicemail number.

AT+STTONE GSM/GPRS Radio Modem-Start or Stop Generating a Tone

AT+STTONE	Start or Stop Generating a Tone
Command Function	This command allows the user to start generating a tone or stop generating a tone.
Syntax Query	AT+STTONE=?
Syntax	+STTONE: (0-1),(1-8,16-18),(0-15300000) OK
Write Syntax	AT+STTONE=<mode>[,<tone>[,<duration>]]
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
< mode >	0 - Stop generating a tone. For stop generating a tone, the AT command is AT+STTONE=0, <tone>. 1 - Start generating a tone.

AT+STTONE	Start or Stop Generating a Tone
< tone >	The value of tone is as follows: 1 - Dial Tone 2 - Called Subscriber Busy 3 - Congestion 4 - Radio Path Acknowledge 5 - Radio path not Available/Call Dropped 6 - Error/Special Information 7 - Call Waiting Tone 8 - Ring Tone 16 - General Beep 17 - Positive Acknowledgement tone 18 - Negative Acknowledgement or Error Tone When the optional tone is not present, default value is 16, which is a general Beep.
< duration >	0-15300000 in milliseconds. When the optional duration is not present, default value is 500ms. When the duration is 0, it plays once. When the duration is 0, all other tones play once except 2 - called subscriber busy, which plays 4 times.
Notes	All tones generated by audio speaker. The tones need to be stopped before originating calls.
Examples	AT+STTONE=1,7,5000 - Generate Call Waiting tone for 5 seconds. AT+STTONE=0,7 - Stop Call Waiting tone.

AT+VTS DTMF and Tone Generation

AT+VTS	DTMF and Tone Generation
Command Function	This command allows the transmission of DTMF tones and arbitrary tones (see note). These tones may be used (for example) when announcing the start of a recording period. The command is write only. In this profile of commands, this command does not operate in data mode of operation.
Syntax Query	AT+VTS=?
Syntax	+VTS: (0-9,#,*,A-D),(1-255) OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+VTS=<DTMF>
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	

AT+VTS	DTMF and Tone Generation
<DTMF>	0 1 2 3 4 5 6 7 8 9 # *
Notes	In GSM this operates only in voice mode. Fixed tone duration. It is expected that the application (Host) will handle the pausing and waits (A,B,C,D)

ATA Answer a Call

ATA	Answer a Call
Command Function	Answers an incoming call.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	ATA
Execute Response	
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Auto answer can be enabled using ATS0.

ATD Dial command

ATD	Dial command
Command Function	This command is used to setup an outbound voice or data call.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	ATD1234567I;
Execute Response	NO DIALTONE or NO CARRIER or CONNECT <value> (for Data only) or BUSY or OK
Unsolicited Response	N/A
Parameter Values	
<n>	V.25ter Dialing Digits = 0 – 9, *, #, +, A, B, C V.25ter Dialing Modifiers = , (comma), T, P, !, @, W
<cmod>	GSM Modifier Characters I = Restrict CLI, i = Allow CLI (Not applicable to HS 1000)

ATD	Dial command
<;>	Semicolon after dialing string or modifier indicates voice call and forces TA into command mode after successful completion.
Notes	Modem Responses: NO DIALTONE - if no dial tone is detected NO CARRIER - if call cannot be set up CONNECT <value> - when connected in a non-voice call (data mode) <value> dependent on ATX setting BUSY - if dialed number is busy OK - when successful voice call or TA ends current call and returns to command mode

ATD> Originate Call Using Phonebook Memory

ATD>	Originate Call Using Phonebook Memory
Command Function	This command is used to setup an outbound voice or data call from a specific phonebook location.
Syntax Query	ATD?
Syntax	NO CARRIER
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	ATD><storage><n><cmod><;>
Execute Response	NO DIALTONE or NO CARRIER or CONNECT <value> or BUSY or OK
Unsolicited Response	N/A
Parameter Values	
<storage>	Phonebook Location
<n>	Storage location number in selected phonebook
<cmod>	GSM Modifier Characters I = Restrict CLI, i = Allow CLI
<;>	Semicolon after dialing string or modifier forces TA into command mode after successful completion.

ATD>	Originate Call Using Phonebook Memory
Notes	Phonebook Location Values: "EN" - SIM (or ME) emergency number "FD" - SIM fixed-dialing-phonebook "LD" - SIM last-dialing-phonebook "BD" - SIM barred-dialing phonebook "SD" - SIM service numbers "LR" - Last received numbers (nonstandard) "AD" - Abbreviated dialing numbers (nonstandard) "LM" - Last missed numbers (nonstandard) "AF" - comb. of fixed and abbrev. dialing phonebook (nonstandard) "SM" - comb. of fixed and abbrev. dialing phonebook (nonstandard) “UD” - User defined Modem Responses: NO DIALTONE - if no dial tone is detected NO CARRIER - if call cannot be set up CONNECT <value> - when connected in a non-voice call (data mode) <value> dependent on ATX setting BUSY - if dialed number is busy OK - when successful voice call or TA ends current call and returns to command mode
Example:	ATD>FD2I The TA will dial the number stored in memory location 2 the fixed-dialing phonebook. The call will block the CLI when made.

ATH Hook Control

ATH	Hook Control
Command Function	Disconnect an existing PAD session.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	ATH
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	If data call or session is active, +++ (escape sequence) must be entered to go to command mode prior to sending ATH command.

ATP Pulse Dialing

ATP	Pulse Dialing
Command Function	Select pulse dialing.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	ATP
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	This command has no affect on GSM.

ATO Return to Data State

ATO	Return to Data State
Command Function	This command issued to return to online mode from command mode when a PAD data call is active.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	ATO
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	N/A

ATS0 Automatic Response to a Network Request for PDP Context Activation

ATS0	Automatic Response to a Network Request for PDP Context Activation
Command Function	The V.25ter 'S0=n' (Automatic answer) command may be used to turn off (n=0) and on (n>0) the automatic response to a network request for a PDP context activation.
Syntax Query	ATS0=?
Syntax	S0:(0-255) OK
Write Syntax	ATS0=<n>
Write Response	OK
Read Syntax	ATS0?
Read Response	000 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<n>	0 = do not answer n>0 = establish data session
Notes	N/A

ATS0 Rings Before Automatic Answer

ATS0	Rings Before Automatic Answer
Command Function	Sets the number of rings before automatically answering a call for GSM and enables automatic answer to a network request for PDP activation.
Syntax Query	ATS0=?
Syntax	S0(0-255) OK
Write Syntax	ATS0=<value>
Write Response	OK
Read Syntax	ATS0?
Read Response	<value> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	N/A
Notes	ATS0=000 will disable auto answer for GSM. If AT+CGAUTO is = to 2 or 3 (default), the MT shall attempt to perform a GPRS attach if it is not already attached, when the 'S0=n' (n>0) command is received. With default settings, if ATS0=(>0) is sent immediately after power up, an error will be returned because the MT will attempt to do an attach before the AT+CREG state has changed to 1.

ATS10 Hang Up Delay

ATS10	Hang Up Delay
Command Function	This command sets the length of time, in tenths of seconds, to wait before disconnecting after the carrier is lost. This is a dummy command that will display a value that has been set, but does not affect functionality.
Syntax Query	ATS10=?
Syntax	S10:(1-254) OK
Write Syntax	ATS10=<value>
Write Response	OK
Read Syntax	ATS10?
Read Response **Response (if HS 1000)**	001 OK 014 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Does not affect GSM functionality.

ATS6 Pause Before Blind Dialing

ATS6	Pause Before Blind Dialing
Command Function	Sets the number of seconds to wait after dialtone detection before dialing. This is a dummy command and does not affect functionality.
Syntax Query	ATS6=?
Syntax	S6:(2-10) OK
Write Syntax	ATS6=<value>
Write Response	OK
Read Syntax	ATS6?
Read Response	002 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Does not affect GSM functionality.

ATS7 Wait for Completion

ATS7	Wait for Completion
Command Function	This command sets the number of seconds to wait after dial tone detection before dialing a number. This is a dummy command that will display a value that has been set, but does not affect functionality.
Syntax Query	ATS7=?
Syntax	S7:(1-255) OK
Write Syntax	ATS7=<value>
Write Response	OK
Read Syntax	ATS7?
Read Response **Response (if HS 1000)**	060 OK 030 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Does not affect GSM functionality.

ATS8 Dial Pause

ATS8	Dial Pause
Command Function	This command sets the number of seconds to wait for the comma dial modifier in the ATD dial string. This is a dummy command that will display a value that has been set, but does not affect functionality.
Syntax Query	ATS8=?
Syntax	S8:(0-255) OK
Write Syntax	ATS8=<value>
Write Response	OK
Read Syntax	ATS8?
Read Response	002 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Does not affect GSM functionality.

ATT Tone Dialing

ATT	Tone Dialing
Command Function	Select tone dialing.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	ATT
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	This command has no effect on GSM.

H Manual Rejection of a Network Request for PDP Context Activation

H	Manual Rejection of a Network Request for PDP Context Activation
Command Function	The V.25ter 'H' or 'H0' (On-hook) command may be used to reject a network request for PDP context activation announced by the unsolicited result code RING.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	H
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	N/A

Network Service Commands

AT%ALS Alternating Line Service

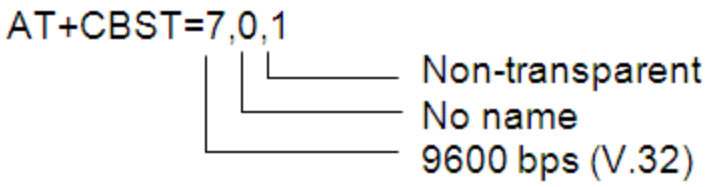
AT%ALS	Alternating Line Service
Command Function	Alternate Line Service provides the MS with the capability of associating two alternate lines with one IMSI. A user will be able to make and receive calls on either line as desired and will be billed separately for calls on each line. Each line will be associated with a separate directory number (MSISDN) and separate subscription profile.
Syntax Query	AT%ALS=?
Syntax	%ALS: (0) OK
Write Syntax	AT%ALS=<line>
Write Response	OK
Read Syntax	AT%ALS?
Read Response	%ALS: 0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<line>	line number 0 = line one
Notes	N/A

AT+CAOC Advice of Charge

AT+CAOC	Advice of Charge
Command Function	This refers to Advice of Charge supplementary service that enables subscriber to get information about the cost of calls. With <mode>=0, the execute command returns the current call meter value from the ME.
Syntax Query	AT+CAOC=?
Syntax	+CAOC: (0-2) OK
Write Syntax	AT+CAOC=<mode>
Write Response	
Read Syntax	AT+CAOC?
Read Response	+CAOC: 1 OK
Execute Syntax	AT+CAOC
Execute Response	+CAOC: “000000” OK
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = Query CCM value 1 = Deactivate 2 = Activate
Notes	When <mode>=0, execution command will return the current call meter value.

AT+CBST Select Bearer Service Type

AT+CBST	Select Bearer Service Type
Command Function	This command is used to select the bearer service with data rate and the connection element to be used when data calls are originated.
Syntax Query	AT+CBST=?
Syntax	+CBST: (0-7, 12, 14, 65, 66, 68, 70, 71,75), (0-1), (0-3)
Write Syntax	AT+CBST=<baud rate>,<name>,<ce>
Write Response	OK/ERROR
Read Syntax	AT+CBST?
Read Response	+CBST: 7,0,1
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	

AT+CBST	Select Bearer Service Type
<baud rate>	0 = autobauding (automatic selection of the speed; this setting is possible in case of 3.1 kHz modem and non-transparent service) 1= 300 bps (V.21) 2 = 1200 bps (V.22) 3 = 1200/75 bps (V.23) 4 = 2400 bps (V.22bis) 5 = 2400 bps (V.26ter) 6 = 4800 bps (V.32) 9600 bps (V.32) 12 = 9600 bps (V.34) 14 = 14400 bps (V.32) 300 bps (V.110) 66 = 1200 bps (V.110) 68 = 2400 bps (V.110 or X.31 flag stuffing) 70 = 4800 bps (V.110 or X.31 flag stuffing) 71 = 9600 bps (V.110 or X.31 flag stuffing) 75 = 14400 bps (V.110 or X.31 flag stuffing)
<name>	0 = data circuit asynchronous (UDI or3.1 kHz modem) 1 = data circuit synchronous (UDI or 3.1 kHz modem)
<ce>	0 = transparent 1 = non-transparent 2 = both, transparent preferred 3 = both, non-transparent preferred
Notes	N/A
Example	

AT+CCFC Call Forwarding Number and Condition

AT+CCFC	Call Forwarding Number and Condition
Command Function	This command allows control of the call forwarding supplementary service. Registration erasure, activation, deactivation, and status query are supported. When querying the status of a network service (<mode> = 2), the response line for “not active” (<status> = 0) should be returned only if service is not active for any <class>.
Syntax Query	AT+CCFC=?
Syntax	+CCFC: (0-5) OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CCFC=<reas>,<mode>[,<number>[,<type>[,<class>[,<time>]]]]
Execute Response	If <mode> <> 2 and command successful OK If <mode> = 2 and command successful (only in connection with <reason> 03) +CCFC: <status>, <class1>[,<number>,<type>[,<time>]] [<CR><LF>+CCFC: ….] OK If error is related to ME functionality: +CME ERROR: <err>
Unsolicited Response	N/A
Parameter Values	

AT+CCFC	Call Forwarding Number and Condition
<reas>	0 = unconditional 1 = mobile busy 2 = no reply 3 = not reachable 4 = all call forwarding 5 = all conditional call forwarding
<mode>	0 = disable 1 = enable 2 = query status 3 = registration 4 = erasure
<number>	string type phone number of forwarding address in format specified by <type>
<type>	type of address in integer format; default 145 when dialing string includes international access code character “+”, otherwise 129
<class>	1 = voice 2 = data 4 = fax (fax not supported) 8 = short message service 16 = data circuit sync 32 = data circuit async
<subaddr>	string type subaddress of format specified by <satype>
<satype>	type of subaddress octet in integer format (refer GSM 04.08 [8] subclause 10.5.4.8); default 128
<time>	time to wait before call is forwarded, rounded to a multiple of 5 sec Default is 20. 1…20..30 (only for <reas>=no reply)

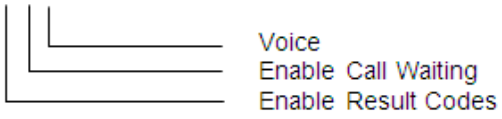
AT+CCFC	Call Forwarding Number and Condition
<status>	0 = not active 1 = active
Notes	
Example:	To call forward all voice calls, no reply after five seconds: AT+CCFC=2,1,"+1719xxxxxx",145,1,,,5 Forward after 5 seconds All Voice calls International number type Forward to this number Enable forwarding No reply

AT+CCUG Closed User Group

AT+CCUG	Closed User Group
Command Function	This command allows control of the Closed User Group supplementary service. Write command with <n>=1 enables to control the CUG information on the air interface as a default adjustment for all following outgoing calls.
Syntax Query	AT+CCUG=?
Syntax	+CCUG: (0, 1), (0,-10), (0-3) OK
Write Syntax	AT+CCUG= [<n> [,<index> [,<info>]]]
Write Response	N/A
Read Syntax	AT+CCUG?
Read Response	+CCUG: 0, 0, 0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<n>	0 = disable CUG temporary mode 1 = enable CUG temporary mode
<index>	0-9 = CUG index 10 = no index (preferred CUG taken from subscriber data)
<info>	0 = no information 1 = suppress OA 2 = suppress preferential CUG 3 = suppress OA and preferential CUG
Notes	N/A

AT+CCWA Call Waiting

AT+CCWA	Call Waiting
Command Function	This command allows control of the Call Waiting supplementary service. Activation and deactivation are supported.
Syntax Query	AT+CCWA=?
Syntax	+CCWA: (0,1) OK
Write Syntax	AT+CCWA=<n>,<mode>,<class>
Write Response	OK
Read Syntax	AT+CCWA?
Read Response	+CCWA: 0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<n>	Sets/shows results code presentation in TA 0 = Disable 1 = Enable
<mode>	0 = Disable 1 = Enable 2 = Query status
<class>	1 = Voice 2 = Data 4 = Fax (fax not supported)
Notes	Not all networks support call waiting for data and fax. Please contact service provider for details.

AT+CCWA	Call Waiting
Example	AT+CCWA=1,1,1 

AT+CCWE Call Meter Maximum Event

AT+CCWE	Call Meter Maximum Event
Command Function	Shortly before the ACM (Accumulated Call Meter) maximum value is reached, an unsolicited result code +CCWV will be sent, if enabled by this command.
Syntax Query	AT+CCWE=?
Syntax	+CCWE: (0,1) OK
Write Syntax	AT+CCWE=<mode>
Write Response	OK
Read Syntax	AT+CCWE?
Read Response	+CCWE: 0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = Disable the call meter warning event 1 = Enable the call meter warning event
Notes	Used in conjunction with AT+CACM, AT+CAOC and AT+CAMM

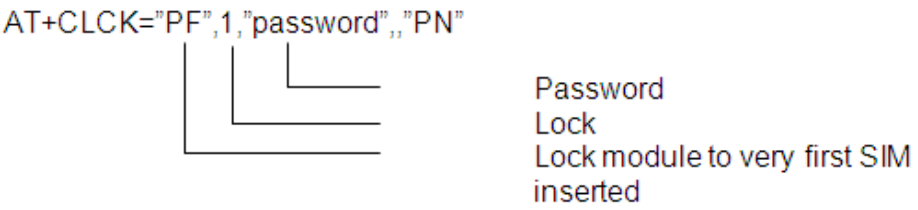
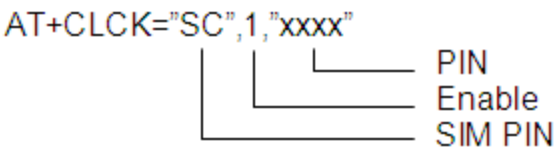
AT+CIMI Request IMSI

AT+CIMI	Request IMSI
Command Function	This command is used to obtain the International Mobile Subscriber Identity (IMSI) value assigned to the SIM.
Syntax Query	AT+CIMI=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CIMI
Execute Response	310260101xxxxx OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Return value is manufacturer specific. The TA returns the International Mobile Subscriber Identity (IMSI).

AT+CLCK Facility Lock

AT+CLCK	Facility Lock
Command Function	This command is used to lock, unlock or interrogate a ME or a network facility <fac>. When querying the status of a network service (<mode>=2) the response line for a “not active” case (<status=0>) should be returned only if service is not active for any <class>. Is should be possible to abort the command when network facilities are set or interrogated.
Syntax Query	AT+CLCK=?
Syntax	+CLCK: ("SC","AO","OI","OX","AI","IR","AB","AG","AC","FD", "PS","PN","PU","PP","PC","PF","FC","FM","MU","MM","AL") OK
Write Syntax	AT+CLCK=<fac>, <mode> [,<passwd> [, <class>]]
Write Response	If <mode><> 2 and command is successful then OK If <mode>=2 and command is successful then +CLCK:<status>,[,<class1>[<CR><LF>+CLCK: <status>, class2…]] OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	

AT+CLCK	Facility Lock
<fac>	“SC” = (SIM PIN 1) “AO” = (Barr All Outgoing Calls) “OI” = (Barr Outgoing International Calls) “OX” = (Barr Outgoing International Calls except Home Country) “AI” = (Barr All Incoming Calls) “IR” = (Barr Incoming Calls when Roaming outside the Home Country) “AB” = (All Barring Services) “AG” = (All Outgoing Barring) “AC” = (All incoming Barring) “FD” = (SIM Fixed Dialing Feature) "PC" = (Corporate Personalization, allows personalization to custom corporate group settings) "PP" = (Provider Personalization, allows for personalization to custom service provider defined groups) "PS" = PH-SIM (lock PHone to SIM card) (ME asks password when other than current SIM card inserted; ME may remember certain amount of previously used cards thus not requiring password when they are inserted) "PN" = Network Personalisation (refer GSM 02.22 [33]) "PU" = network sUbset Personalisation (refer GSM 02.22 [33]) "PF" = lock Phone to the very First inserted SIM card (also referred in the present document as PH-FSIM) (ME asks password when other than the first SIM card is inserted) "FC" = Fail count "FM" = Fail count reset "MU" = Master unlock "MM" = Master unlock menu "AL" = alternating Line service (PIN2)

AT+CLCK	Facility Lock
<mode>	0 = Unlock 1 = Lock 2 = Query Status
<passwd>	“password”
<class>	1 = voice 2 = data 4 = fax (fax not supported) 7 = all classes (default) 8 = short message service
<status>	0 = Off 1 = On
Notes	Quotation marks are optional when entering <passwd>.
Example:	To set Network Personalization on first SIM inserted  To enable SIM PIN 

AT+CLIP Calling Line Identification Presentation

AT+CLIP	Calling Line Identification Presentation
Command Function	This command refers to the GSM supplementary service CLIP (Calling Line Identification Presentation) that enables a called subscriber to get the Calling Line Identity (CLI) of the calling party when receiving a mobile terminated call. The write command enables or disables the presentation of the CLI at the TE. It has no effect on the execution of the supplementary service CLIP in the network.
Syntax Query	AT+CLIP=?
Syntax	+CLIP: (0,1) OK
Write Syntax	AT+CLIP=<n>
Write Response	+CLIP: <n> or OK or +CME ERROR: <err>
Read Syntax	AT+CLIP?
Read Response	+CLIP: <n>, <m> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Unsolicited Result Code	When CLIP is enabled at the TE (and is permitted by the calling subscriber), an unsolicited result code is returned after every RING (or +CRING: <type>) at a mobile terminating call
Voice call response format:	+CLIP: <number>, <type>,,,,<CLI validity> Data/FAX call response format: +CLIP: <number>, <type>
Parameter Values	

AT+CLIP	Calling Line Identification Presentation
<n>	0 = suppress unsolicited results codes 1 = display unsolicited result codes
<m>	0 = CLIP not enabled 1 = CLIP enabled 2 = Unknown
<number>	string type phone number of calling address in format specified by <type>
<type>	type of address octet in integer format: 145 when dialing string includes international access code character “+”, otherwise 129
<CLI validity>	0 = CLI valid 1 = CLI has been withheld by the originator 3 = CLI is not available due to inter-working problems or limitations of originating network. <number> shall be an empty string (“”) and <type> value will not be significant.
Notes	N/A

AT+CLIR Calling Line Identification

AT+CLIR	Calling Line Identification Restriction
Command Function	This command allows a calling subscriber to enable or disable the presentation of the CLI to the called party when originating a call. The write command overrides the CLIR subscription (default is restricted or allowed) when temporary mode is provisioned as a default adjustment for all outgoing calls. This adjustment can be revoked by using the opposite command. This command, when used by a subscriber, without provision of CLIR in permanent mode the network will act according GSM 02.81 [3]. The read command gives the default adjustment for all outgoing calls (given in <n>), and also triggers and interrogation of the provision status of the CLIR service (given in <m>).
Syntax Query	AT+CLIR=?
Syntax	+CLIR: (0,1,2) OK
Write Syntax	AT+CLIR=[<n>]
Write Response	N/A
Read Syntax	AT+CLIR?
Read Response	+CLIR: <n>, <m> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<n>	(parameter sets the adjustment for outgoing calls) 0 = presentation indicator is used according to the subscription of the CLIR service 1 = CLIR Invocation 2 = CLIR suppression

AT+CLIR	Calling Line Identification Restriction
<m>	(parameter shows the subscriber CLIR service status in the network) 0 = CLIR not enabled 1 = CLIR enabled in permanent mode 2 = Unknown (e.g. no network, etc.) 3 = CLIR temporary mode presentation restricted 4 = CLIR temporary mode presentation allowed
Notes	N/A

AT+CMER Mobile Termination Event Reporting

AT+CMER	Mobile Termination Event Reporting
Command Function	Set command enables or disables sending of unsolicited result codes from TA to TE in the case of key pressings, display changes, and indicator state changes. <mode> controls the processing of unsolicited result codes specified within this command. <bfr> controls the effect on buffered codes when <mode> 1, 2 or 3 is entered. If setting is not supported by the MT, +CME ERROR: <err> is returned. Refer subclause 9.2 for <err> values. Test command returns the modes supported as compound values.
Syntax Query	AT+CMER=?
Syntax	+CMER: (0-2), (0), (0), (0-2), (0,1)
Write Syntax	AT+CMER=[<mode>[,<keyp>[,<disp>[,<ind>[,<bfr>]]]]]
Write Response	OK
Read Syntax	AT+CMER?
Read Response	+CMER:<mode>,<keyp>, <disp>,<ind>,<bfr>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	

AT+CMER	Mobile Termination Event Reporting
<mode>	buffer unsolicited result codes in the TA; if TA result code buffer is full, codes can be buffered in some other place or the oldest ones can be discarded discard unsolicited result codes when TA-TE link is reserved (e.g. in on-line data mode); otherwise forward them directly to the TE buffer unsolicited result codes in the TA when TA-TE link is reserved (e.g. in on-line data mode) and flush them to the TE after reservation; otherwise forward them directly to the TE forward unsolicited result codes directly to the TE; TA-TE link specific inband technique used to embed result codes and data when TA is in on-line data mode
<keyp>	no keypad event reporting keypad event reporting using result code +CKEV: <key>,<press>. <key> indicates the key (refer IRA values defined in table in subclause "Keypad control +CKPD") and <press> if the key is pressed or released (1 for pressing and 0 for releasing). Only those key pressings, which are not caused by +CKPD shall be indicated by the TA to the TE. NOTE 1:When this mode is enabled, corresponding result codes of all keys currently pressed should be flushed to the TA regardless of <bfr> setting. keypad event reporting using result code +CKEV: <key>,<press>. All key pressings shall be directed from TA to TE. NOTE 2:When this mode is enabled, corresponding result codes of all keys currently pressed should be flushed to the TA regardless of <bfr> setting.
<disp>	no display event reporting display event reporting using result code +CDEV: <elem>,<text>. <elem> indicates the element order number (as specified for +CDIS) and <text> is the new value of text element. Only those display events, which are not caused by +CDIS shall be indicated by the TA to the TE. Character set used in <text> is as specified by command Select TE Character Set +CSCS display event reporting using result code +CDEV: <elem>,<text>. All display events shall be directed from TA to TE. Character set used in <text> is as specified by command Select TE Character Set +CSCS

AT+CMER	Mobile Termination Event Reporting
<ind>	no indicator event reporting indicator event reporting using result code +CIEV: <ind>,<value>. <ind> indicates the indicator order number (as specified for +CIND) and <value> is the new value of indicator. Only those indicator events, which are not caused by +CIND shall be indicated by the TA to the TE indicator event reporting using result code +CIEV: <ind>,<value>. All indicator events shall be directed from TA to TE
<bfr>	TA buffer of unsolicited result codes defined within this command is cleared when <mode> 1...3 is entered TA buffer of unsolicited result codes defined within this command is flushed to the TE when <mode> 1...3 is entered (OK response shall be given before flushing the codes)
Notes	N/A

AT+CMOD Call mode

AT+CMOD	Call mode
Command Function	This command is used to select the type of call mode desired for following dial (D) and/or answer (A) commands.
Syntax Query	AT+CMOD=?
Syntax	+CMOD: (0-3) OK
Write Syntax	AT+CMOD=<mode>
Write Response	OK
Read Syntax	AT+CMOD?
Read Response	+CMOD: 0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	0 - Single service 1 - Alternating voice/fax (teleservice 61) 2 - Alternating voice/data (bearer service 61) 3 - Voice followed by data (bearer service 81)
Notes	Default value will be 0. AT&F, restore factory defaults will reset this value to 0.

AT+CNUM Subscriber Number

AT+CNUM	Subscriber Number
Command Function	This command is used to obtain the MSISDNs related to the subscriber.
Syntax Query	AT+CNUM=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CNUM
Execute Response	+CNUM: “Line1”, “xxx xxx xxxx”, 145 OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	

AT+COLP Connected Line Identification

AT+COLP	Connected Line Identification Presentation
Command Function	This command is enables a calling subscriber to get the Connected Line Identity (COL) of the called party after setting up a mobile originated call. The command enables or disables the presentation of the COL at the TE. It has no effect on the execution of the supplementary service COLR in the network.
Syntax Query	AT+COLP=?
Syntax	+COLP: (0,1) OK
Write Syntax	AT+COLP= [<n>]
Write Response	OK
Read Syntax	AT+COLP?
Read Response	+COLP: <n>,<m> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<n>	(parameter sets/shows the result code presentation status in the TA) 0 = disable 1 = enable
<m>	(parameter shows the subscriber COLP) 0 = COLP not enabled 1 = COLP enabled 2 = Unknown (e.g. no network, etc.)
Notes	N/A

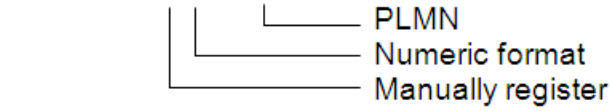
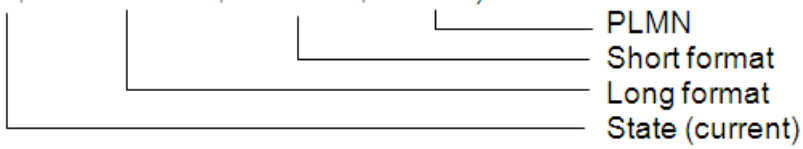
AT+COPN Read Operator Names

AT+COPN	Read Operator Names
Command Function	Execute command returns the list of operator names from the ME.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+COPN
Execute Response	+COPN: <numeric1>,<alpha1>[<CR><LF>+COPN: <numeric2>, <alpha2>[...]] OK
Unsolicited Response	N/A
Parameter Values	
<numericn>	string type; operator in numeric format (see +COPS)
<alphan>	string type; operator in long alphanumeric format (see +COPS)
Notes	N/A

AT+COPS Operator Selection

AT+COPS	Operator Selection
Command Function	Write command forces an attempt to select and register the GSM network operator. <mode> is used to select whether the selection is done automatically by the ME or is forced by this command to operator <oper> (it shall be given in format <format>). If the selected operator is not available, no other operator shall be selected (except <mode> = 4). The selected operator name format shall apply to further read commands (+COPS?) also. <mode>=2 forces an attempt to deregister from the network. The selected mode affects to all further registration (e.g. after <mode>=2, ME shall be unregistered until <mode>=0 or 1 is selected). Read command returns the current mode and the currently selected operator. If no operator is selected, <format> and <oper> are omitted. Test command returns a list of quadruplets, each representing an operator present in the network. Quadruplet consists of an integer indicating the availability of the operator <stat>, long and short alphanumeric format of the name of the operator, and numeric format representation of the operator. Any of the formats may be unavailable and will then be an empty field (,,). The list of operators comes in the following order: Home network, networks referenced in SIM, and other networks.
Syntax Query	AT+COPS=?
Syntax	+COPS: (2, " ", " ", "31022"), (3, " "," ", "310380") OK
Write Syntax	AT+COPS=<mode>
Write Response	[, <format> [, oper>]] OK or +CME ERROR: <err>
Read Syntax	AT+COPS?
Read Response	+COPS: 0 OK
Execute Syntax	N/A
Execute Response	N/A

AT+COPS	Operator Selection
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = automatic (<oper> field is ignored) 1 =manual (<oper> field shall be present) 2 = deregister from network 3 = set only <format> (for read command +COPS?), do not attempt registration/deregistration (<oper> field is ignored); this value is not applicable in read command response 4 = manual/automatic (<oper> field shall be present); if manual selection fails, automatic mode (<mode=0) is entered
<format>	0 = long format alphanumeric <oper> 1 = short format alphanumeric <oper> 2 = numeric <oper>; GSM Location Area Identification Number
<oper>	operator in format as in per <format>
<stat>	0 = Unknown 1 = Available 2 = Current 3 = Forbidden
Notes	

AT+COPS	Operator Selection
Example	To manually register the modem on a known PLMN: AT+COPS=1,2,"xxxxx"  To read operator information: AT+COPS=? +COPS: (2,"Voicestream","Vstream","31022") 

AT+CPOL Preferred Operator List

AT+CPOL	Preferred Operator List
Command Function	This command is used to list and edit the SIM preferred list of networks.
Syntax Query	AT+CPOL=?
Syntax	+CPOL: (1-n), (0-2) OK
Write Syntax	AT CPOL=[<index>][, <format>[,<oper>]]
Write Response	OK
Read Syntax	AT+CPOL?
Read Response	+CPOL: <index1>,<format>,<oper1>… <index10>,<format>,<oper10> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<indexn>	integer type; the order number of operator in the SIM preferred operator list
<format>	0 = long format alphanumeric <oper> 1 = short format alphanumeric <oper> 2 = numeric <oper>
<opern>	string type; <format> indicates if the format is alphanumeric or numeric (see +COPS)
Notes	This command is used to edit the SIM preferred list of networks. Execute command writes an entry in the SIM list of preferred operators (EFPLMNsel). If <index> is given but <oper> is left out, entry is deleted. If <oper> is given but <index> is left out, <oper> is put in the next free location. If only <format> is given, the format of the <oper> in the read command is changed.

AT+CRLP Radio Link Protocol Parameters

AT+CRLP	Radio Link Protocol Parameters
Command Function	This command is used to select the radio link protocol parameters.
Syntax Query	AT+CRLP=?
Syntax	+CRLP: (0-61), (0-61), (39-255), (1-255) OK
Write Syntax	AT+CRLP=<iws>,<mws>,<T1>,<N2>
Write Response	OK/ERROR
Read Syntax	AT+CRLP?
Read Response	+CRLP: 61, 61, 48, 6 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<iws>	IWF to MS window size values = 0 to 61 (61 recommended)
<mws>	MS to IWF window size values = 0 to 61 (61 recommended)
<T1>	Acknowledgement timer values = halfrate >380ms (480 recommended) fullrate >600ms (780 recommended)
<N2>	Retransmission attempts values = >0 (6 recommended)
Notes	N/A

AT+CSNS Single Numbering Scheme

AT+CSNS	Single Numbering Scheme
Command Function	This command selects the bearer or teleservice to be used when mobile terminated single numbering scheme call is established. Parameter values set with +CBST command shall be used when <mode> equals to a data service. If +CBST parameter is set to a value that is not applicable to single numbering calls, ME/TA shall map the value to the closest valid one. E.g. if user has set <speed>=71, <name>=0 and <ce>=1 (non-transparent asynchronous 9600 bps V.110 ISDN connection) for mobile originated calls, ME/TA shall map the values into non-transparent asynchronous 9600 bps V.32 modem connection when single numbering scheme call is answered.
Syntax Query	AT+CSNS=?
Syntax	+CSNS: (0-7) OK
Write Syntax	AT+CSNS = <mode>
Write Response	OK
Read Syntax	AT+CSNS?
Read Response	+CSNS: 0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	

AT+CSNS	Single Numbering Scheme
<mode>	0 = voice 1 = alternating voice/fax, voice first (TS 61) 2 = fax (TS 62) 3= alternating voice/data, voice first (BS 61) 4 = data 5 = alternating voice/fax, fax first (TS 61) 6 = alternating voice/data, data first (BS 61) 7 = voice followed by data (BS 81)
Notes	Fax not supported

Supplementary Service Commands

AT+CSSN Supplementary Service Notifications

AT+CSSN	Supplementary Service Notifications
Command Function	This command refers to supplementary service related network initiated notifications. The set command enables/disables the presentation of notification result codes from TA to TE. When <n>=1 and a supplementary service notification is received after a mobile originated call setup, intermediate result code +CSSI: <code1>[,<index>] is sent to TE before any other MO call setup result codes are presented. When several different <code1>s are received from the network, each of them shall have its own +CSSI result code. When <m>=1 and a supplementary service notification is received during a mobile terminated call setup or during a call; or when a forward check supplementary service notification is received, unsolicited result code +CSSU: <code2>[,<index>[,<number>,<type>[,<subaddr>,<satype>]]] is sent to TE. In case of MT call setup, result code is sent after every +CLIP result code (refer command "Calling line identification presentation +CLIP"). When several different <code2>s are received from the network, each of them shall have its own +CSSU result code.
Syntax Query	AT+CSSN=?
Syntax	+CSSN: (0,1),(0,1) OK
Write Syntax	AT+CSSN=<n>,<m>
Write Response	OK
Read Syntax	AT+CSSN?
Read Response	+CSSN: <n>,<m> OK
Execute Syntax	N/A

AT+CSSN	Supplementary Service Notifications
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<n>	(parameter sets/shows the +CSSI result code presentation status in the TA): 0 = disable 1 = enable
<m>	(parameter sets/shows the +CSSU result code presentation status in the TA): 0 = disable 1 = enable
<code1>	0 = unconditional call forwarding is active 1 = some of the conditional call forwardings are active 2 = call has been forwarded 3 = call is waiting 4 = this is a CUG call (also <index> present) 5 = outgoing calls are barred 6 = incoming calls are barred 7 = CLIR suppression rejected 8 = call has been deflected
<index>	refer "Closed user group +CCUG"

AT+CSSN	Supplementary Service Notifications
<code2>	0 = this is a forwarded call (MT call setup) 1 = this is a CUG call (also <index> present) (MT call setup) 2 = call has been put on hold (during a voice call) 3 = call has been retrieved (during a voice call) 4 = multiparty call entered (during a voice call) 5 = call on hold has been released (this is not a SS notification) (during a voice call) 6 = forward check SS message received (can be received whenever) 7 -= call is being connected (alerting) with the remote party in alerting state in explicit call transfer operation (during a voice call) 8 = call has been connected with the other remote party in explicit call transfer operation (also number and subaddress parameters may be present) (during a voice call or MT call setup) 9 = this is a deflected call (MT call setup)
<number>	string type phone number of format specified by <type>
<type>	type of address octet in integer format
<subaddr>	string type subaddress of format specified by <satype>
<satype>	type of subaddress octet in integer format
Notes	N/A

AT+CUSD Unstructured Supplementary Service

AT+CUSD	Unstructured Supplementary Service
Command Function	This command allows control of the Unstructured Supplementary Service Data (USSD)]. Both network and mobile initiated operations are supported. Parameter <n> is used to disable/enable the presentation of an unsolicited result code (network initiated operation) to the TE.
Syntax Query	AT+CUSD=?
Syntax	+CUSD: (0,1,2) OK
Write Syntax	+CUSD=[<n>[,<str>[,<dcs>]]]
Write Response	OK
Read Syntax	AT+CUSD?
Read Response	+CUSD: 0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<n>	0 = disable the result code presentation 1 = enable the result code presentation 2 = cancel session
<str>	(when <str> parameter is not given, network is not interrogated) actual USSD string in “quotes”
<dcs>	language parameter see GSM 03.38 - Default 15 (Language unspecified)
Notes	

AT+CUSD	Unstructured Supplementary Service
Example	AT+CUSD=1,"*201*35#",15 OK +CUSD: 0,"*201*35#",15 (network response) USSD stings can also be sent using the ATD command. ATD*201*35# OK +CUSD: 0,"*201*35#",15 (network response)

SMS Commands

AT+CMGC Send Command

AT+CMGC	Send Command
Command Function	Execution command sends a command message from a TE to the network (SMS-COMMAND). The entering of PDU is done similarly as specified in command Send Message +CMGS. Message reference value <mr> is returned to the TE on successful message delivery
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CMGC=<length>
Execute Response	PDU is given<ctrl-Z> +CMGC: <mr>[,<ackpdu>] OK
Unsolicited Response	N/A
Parameter Values	
<length>	length of PDU message in octets
<mr>	Message reference
<ackpdu>	data element of ack-pdu
Notes	This command only applicable to pdu mode .

AT+CMGD Delete Message

AT+CMGD	Delete Message
Command Function	Deletes message from preferred storage location.
Syntax Query	AT+CMGD=?
Syntax	+CMGD:(0-255),(0-4)
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CMGD=<index>,<status>
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	
<index>	Integer value of memory location.
<status>	0 = Delete the messages specified by the index 1 = Ignore the index and delete all the read messages 2 = Ignore the index and delete all the read and sent messages 3 = Ignore the index and delete all the read, sent and unsent messages. 4 = Ignore the index and delete all the messages whatever the status is
Notes	If there is no message stored in the selected index, an error will be returned.

AT+CMGF SMS Format

AT+CMGF	SMS Format
Command Function	Set command tells the TA, which input and output format of messages to use. <mode> indicates the format of messages used with send, list, read and write commands and unsolicited result codes resulting from received messages. Mode can be either PDU mode (entire TP data units used) or text mode (headers and body of the messages given as separate parameters).
Syntax Query	AT+CMGF=?
Syntax	AT+CMGF: (0,1) OK
Write Syntax	AT+CMGF=<mode>
Write Response	OK
Read Syntax	AT+CMGF?
Read Response	+CMGF: 1 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = PDU mode 1 = Text mode
Notes	Use of PDU mode requires an in depth understanding of PDU message and header formats.

AT+CMGL List Messages

AT+CMGL	List Messages
Command Function	List messages from storage.
Syntax Query	AT+CMGL=?
Syntax	+CMGL: (“REC UNREAD”,”REC READ”,”STO UNSENT”,”STO SENT”,”ALL”) OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CMGL =<stat>
Execute Response	+CMGL: <index>,<stat>,<da/oa>, [<alpha>,<scts>,<tooa/toda>,<length>] <CR><LF> data OK
Unsolicited Response	N/A
Parameter Values	See Notes
<index>	Memory location integer
<stat>	Status of message “REC UNREAD” “REC READ” “STO UNREAD” “STO READ” “ALL”
<do/oa>	destination address
<alpha>	alphanumeric representation of <da> or <oa> corresponding to the entry found in MT phonebook
<scts>	Service center time stamp

AT+CMGL	List Messages
<tooa/toda>	Address Type-of-Address octet in integer format
<length>	Length of message in octets
Notes	Above settings for <stat> assume AT+CMGF=1 (text mode). For AT+CMGF=0 (PDU mode), the following <stat> values are supported: 0,1,2,3,4. Parameters in [] may or may not be reported dependent upon the setting of AT+CMGF. 0 = “Rec Unread” 1 = “Rec Read” 2 = “Sto Unsent” 3 = “Sto Sent” 4 = “ALL”

AT+CMGR Read Message

AT+CMGR	Read Message
Command Function	Read stored messages.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CMGR=<index>
Execute Response	+CMGR: <stat>,<oa>,<scts>,[<tooa>,<fo>,<pid>,<sca>,<tosca>,<length>]<CR><LF><data> OK
Unsolicited Response	N/A
Parameter Values	
<stat>	Status of message (Rec Read, Rec Unread, Sto Unsent, Sto Sent)
<oa>	Originating address
<scts>	Service center time stamp
<tooa>	Originating address – type of address
<fo>	First octet
<pid>	Protocol identifier
<sca>	Service center address
<tosca>	Type of address
<length>	Length of message in octets
Notes	The above parameters are for text mode.

AT+CMGS Send Message

AT+CMGS	Send Message
Command Function	Sends message from the TE to the network.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CMGS=”<da>”,[<toda>]
Execute Response	Enter text <cntl Z> +CMGS <mr> OK
Unsolicited Response	N/A
Parameter Values	
<da>	Destination address
<mr>	Message reference
Notes	The example provided is for text mode (AT+CMGF=1). An in depth understanding of PDU messages is required for PDU mode.

AT+CMGW Write Message to Memory

AT+CMGW	Write Message to Memory
Command Function	Writes message to preferred storage location.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CMGW=<”da”><CR><LF>
Execute Response	Text is entered<cntlZ> +CMGW: <index> OK
Unsolicited Response	N/A
Parameter Values	
<da>	Destination Address
<index>	Integer value of memory location of the stored message
Notes	The above is for text mode only.

AT+CMSS Send Message from Storage

AT+CMSS	Send Message from Storage
Command Function	Sends message (with location value) from preferred message storage.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CMSS=<index>
Execute Response	+CMSS: <mr> OK
Unsolicited Response	N/A
Parameter Values	
<index>	Integer value of location number supported by associated memory
<mr>	Message reference
Notes	The above is for text mode only.

AT+CNMI New Message Indication to TE

AT+CNMI	New Message Indication to TE
Command Function	Selects how incoming messages from the network are indicated to the TE when the TE is active.
Syntax Query	AT+CNMI=?
Syntax	+CNMI: (0-2),(0-3),(0,2),(0,1),(0,1) OK
Write Syntax	AT+CNMI=<mode>,<mt>,<bm>,<ds>,<bfr>
Write Response	OK
Read Syntax	AT+CNMI?
Read Response	+CNMI: 1,1,0,0,0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = Buffer unsolicited result codes in the TA 1 = Discard indication and reject new received message unsolicited result codes when TA-TE link is reserved 2 = Buffer unsolicited result codes in the TA when TA-TE link is reserved and flush them to the TE after reservation
<mt>	Receiving procedure for different message data coding schemes (refer GSM 03.38 [2])

AT+CNMI	New Message Indication to TE
<mt> 0	no class: as in GSM 03.38, but use <mem3> as preferred memory class 0: as in GSM 03.38, but use <mem3> as preferred memory if message is tried to be stored class 1: as in GSM 03.38, but use <mem3> as preferred memory class 2: as in GSM 03.38 class 3: as in GSM 03.38, but use <mem3> as preferred memory message waiting indication group (discard message): as in GSM 03.38, but use <mem3> as preferred memory if message is tried to be stored message waiting indication group (store message): as in GSM 03.38, but use <mem3> as preferred memory
<mt>1	as <mt>=0 but send indication if message stored successfully
<mt>2	no class: route message to TE class 0: as in GSM 03.38, but also route message to TE and do not try to store it in memory class 1: route message to TE class 2: as <mt>=1 class 3: route message to TE message waiting indication group (discard message): as in GSM 03.38, but also route message to TE and do not try to store it in memory message waiting indication group (store message): as <mt>=1
<mt>3	class 3: route message to TE others: as <mt>=1

AT+CNMI	New Message Indication to TE
<bm>	0 <bm> No CBM indications are routed to the TE If CBM is stored into ME/TA, indication of the memory location is routed to the TE using unsolicited result code: +CBMI: <mem>,<index> 1 New CBMs are routed directly to the TE using unsolicited result code 2 Class 3 CBMs are routed directly to TE using unsolicited result codes defined in <bm>=2. If CBM storage is supported, messages of other classes result in indication as defined in <bm>=1
<ds>	0 = No SMS-STATUS_REPORTs are routed to the TE 1 = SMS-STATUS-REPORTs are routed to the TE using unsolicited result code.
<bfr>	0 =TA buffer of unsolicited result codes defined within this command is flushed to the TE when <mode> 1...2 is entered. 1 = TA buffer of unsolicited result codes defined within this command is cleared when <mode> 1...2 is entered.
Notes	N/A

AT+CPMS Preferred Message Storage

AT+CPMS	Preferred Message Storage
Command Function	Set command selects memory storages <mem1>, <mem2> and <mem3> to be used for reading, writing, etc.
Syntax Query	AT+CPMS=?
Syntax	+CPMS: (“ME”,“SM”),(“ME”,“SM”),(“ME”,“SM”) OK
Write Syntax	AT+CPMS=<mem1>,<mem2>,<mem3>
Write Response	+CPMS: 0,25,0,25,0,25 OK
Read Syntax	AT+CPMS?
Read Response	+CPMS: “SM”,0,25,“SM”,0,25,“SM”,0,25 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mem1>	String type; memory from which messages are read and deleted (commands List Messages +CMGL, Read Message +CMGR and Delete Message +CMGD); defined values: "ME" = ME message storage "SM" = SIM message storage
<mem2>	String type; memory to which writing and sending operations are made (commands Send Message from Storage +CMSS and Write Message to Memory +CMGW)); refer to <mem1> for defined values
<mem3>	String type; memory to which received messages are preferred to be stored (unless class of message defines a specific storage location; refer to command New Message Indications +CNMI); refer to<mem1> for defined values
Notes	ME can only store up to three (3) short messages.

AT+CSCA Service Center Address

AT+CSCA	Service Center Address
Command Function	Set command updates the SMSC address, through which mobile originated SMs are transmitted.
Syntax Query	AT+CSCA=?
Syntax	OK
Write Syntax	AT+CSCA=<”sca”>,<tosca>
Write Response	+CSCA: <”sca”>,<tosca> OK
Read Syntax	AT+CSCA?
Read Response	+CSCA=”12063130004”,145 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<”sca”>	SMSC Address
<tosca>	SC address Type-of-Address
Notes	The service center address must be present to complete delivery of SMS. Most SIMs are delivered from the service provider with a service center already programmed into the SIM. A “+” should be entered in front of the sms address, but is not required by all operators.

AT+CSCB Select Cell Broadcast Message Types

AT+CSCB	Select Cell Broadcast Message Types
Command Function	Select which types of CBm's are to be received by the ME.
Syntax Query	AT+CSCB=?
Syntax	+CSCB: (0,1) OK
Write Syntax	AT+CSCB=<mode>
Write Response	OK
Read Syntax	AT+CSCB?
Read Response	+CSCB: 0,"<mids>","<dcss> " OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = Message types specified in <MIDS> and <DCCS> are accepted 1 = Message types specified in <MIDS> and <DCCS> are not accepted
<mids>	string type; all different possible combinations of CBM message identifiers (refer <mid>) (default is empty string); e.g. "0,1,5,320-478,922"
<dcss>	string type; all different possible combinations of CBM data coding schemes (refer <dcs>) (default is empty string); e.g. "0-3,5"
Notes	An understanding of CBM message identifiers and CBM loading schemes is required to properly implement this command. Used in conjunction with AT+CNMI.

AT+CSDH Show Text Mode Parameters

AT+CSDH	Show Text Mode Parameters
Command Function	Determines if detail information is shown in result codes.
Syntax Query	AT+CSDH=?
Syntax	+CSDH: (0,1) OK
Write Syntax	AT+CSDH=<show>
Write Response	OK
Read Syntax	AT+CSDH?
Read Response	+CSDH: 1 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<show>	0 = Do not show header values 1= Show the values in result codes
Notes	N/A

AT+CSMP Set Text Mode Parameters

AT+CSMP	Set Text Mode Parameters
Command Function	Selects additional values needed when the SIM is sent to the network or placed in storage.
Syntax Query	AT+CSMP=?
Syntax	+CSMP: (0-255),(0-255),(0-255),(0-255) OK
Write Syntax	AT+CSMP=<fo>,<vp>,<pid>,<dcs>
Write Response	OK
Read Syntax	AT+CSMP?
Read Response	+CSMP: 17, 167, 0, 0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<fo>	depending on the command or result code: first octet of GSM 03.40 SMS-DELIVER, SMS-SUBMIT (default 17), or SMS-COMMAND (default 2) in integer format
<vp>	depending on SMS-SUBMIT <fo> setting: GSM 03.40 TP-Validity-Period either in integer format (default 167)), in time-string format (refer <dt>), or if is supported, in enhanced format (hexadecimal coded string with quotes)
<pid>	Protocol-Identifier in integer format (default 0), refer GSM 03.40
<dcs>	SMS Data Coding Scheme (default 0), or Cell Broadcast Data Coding Scheme in integer format depending on the command or result code: GSM 03.38
Notes	N/A

AT+CSMS Select Message Service

AT+CSMS	Select Message Service
Command Function	Set command selects messaging service <service>. It returns the types of messages supported by the ME: <mt> for mobile terminated messages, <mo> for mobile originated messages and <bm> for broadcast type messages.
Syntax Query	AT+CSMS=?
Syntax	+CSMS: (0,1) OK
Write Syntax	AT+CSMS=<service>
Write Response	+CSMS: 0,1,1,1 OK
Read Syntax	AT+CSMS?
Read Response	+CSMS: 0,1,1,1 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<service>	0 = Phase 2 version 1 = Phase 2+ version
Notes	N/A

GPRS Commands

AT+CGREG GPRS Network Registration Status

AT+CGREG	GPRS Network Registration Status
Command Function	Controls the presentation of an unsolicited result code +CGREG.
Syntax Query	AT+CGREG=?
Response	+CGREG: (0,2) OK
Write Syntax	AT+CGREG=1
Response	OK
Read Syntax	AT+CGREG?
Response	+CGREG: <n>,<stat>[,<lac>,<ci>] OK
Execute Syntax	N/A
Response	N/A
Parameter Values	
<n>	0 = disable network registration unsolicited result code 1 = enable network registration unsolicited result code +CGREG: <stat> 2 = enable network registration and location information unsolicited result code +CGREG: <stat>[,<lac>,<ci>]

AT+CGREG	GPRS Network Registration Status
<stat>	0 = not registered, ME is not currently searching a new operator to register to 1 = registered, home network 2 = not registered, but ME is currently searching a new operator to register to 3 = registration denied 4 = unknown 5 = registered, roaming
<lac>	String type; two-byte location area code in hexadecimal format (e.g. "00C3" equals 195 in decimal)
<ci>	String type; two-byte cell ID in hexadecimal format
Notes	If parameter <n> is omitted the command does nothing.
Examples	

AT+CGACT PDP Context Activate or Deactivate

AT+CGACT	PDP Context Activate or Deactivate
Command Function	The execution command is used to activate or deactivate the specified PDP context (s).
Syntax Query	AT+CGACT=?
Syntax	+:CGACT: (0,1) OK
Write Syntax	AT+CGACT=<state>,<cid>
Write Response	OK
Read Syntax	AT+CGACT?
Read Response	+CGACT: 1,0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<state>	0 = deactivated 1 = activated
<cid>	numeric value of PDP context activation
Notes	It is not possible to omit the parameter <state>. AT+CGDCONT command must be entered prior to context activation. Activating a PDP Context through AT+CGACT will not allow for the modem to be used in a DUN connection. To use the modem in a DUN connection, use the AT$AREG=2 command instead.

AT+CGANS Manual Response to a Network Request for PDP Context Activation

AT+CGANS	Manual Response to a Network Request for PDP Context Activation
Command Function	The execution command requests the MT to respond to a network request for GPRS PDP context activation which has been signaled to the TE by the RING or +CRING: unsolicited result code.
Syntax Query	AT+CGANS=?
Syntax	+CGANS: (0,1),"PPP" OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CGANS+<Execute Response>,<L2P>
Unsolicited Response	N/A
Execute Response	OK
Unsolicited Response	NA
Parameter Values	
<response>	0 = request is rejected 1 = request is accepted
<L2P>	"PPP"
Notes	Supported value for <L2P>: "PPP".

AT+CGATT GPRS Attach or Detach

AT+CGATT	GPRS Attach or Detach
Command Function	The execution command is used to attach the MT to, or detach the MT from GPRS service.
Syntax Query	AT+CGATT=?
Syntax	+CGATT: (0,1) OK
Write Syntax	AT+CGATT=<state>
Write Response	OK
Read Syntax	AT+CGATT?
Read Response	+GCATT: 0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<state>	0 = detached 1 = attached
Notes	If parameter <state>is omitted the GPRS attach state will be changed.

AT+CGAUTO Automatic Response to a Network Request for PDP Context Activation

AT+CGAUTO	Automatic Response to a Network Request for PDP Context Activation
Command Function	The set command disables or enables an automatic positive response (auto-answer) to the receipt of a Request PDP Context Activation message from the network.
Syntax Query	AT+CGAUTO=?
Syntax	+CGAUTO: (0-3) OK
Write Syntax	AT+CGAUTO=<n>
Write Response	OK
Read Syntax	AT+CGAUTO?
Read Response	+CGAUTO: 3 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<n>	0 = turn off automatic response for GPRS only 1 = turn on automatic response for GPRS only 2 = modem compatibility mode, GPRS only 3 = modem compatibility mode, GPRS and circuit switched calls (default)
Notes	If parameter <n> is omitted it is assumed to be 3 (modem compatibility mode, GPRS and circuit switched calls).

AT+CGCLASS GPRS Mobile Station Class

AT+CGCLASS	GPRS Mobile Station Class
Command Function	Sets the MT to operate to a specified GPRS mobile class.
Syntax Query	AT+CGCLASS=?
Syntax	+CGCLASS: (“B”,”CG”,”CC”) OK
Write Syntax	AT+CGCLASS=<class>
Write Response	OK
Read Syntax	AT+CGCLASS?
Read Response	+CGCLASS: “B” OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<class>	“B” = class B “CG” = class C in GPRS only mode “CC” = class C in circuit switched only mode (lowest)
Notes	If parameter <class> is omitted, a detached mobile attaches with the last class or the default class (“B”).

AT+CGDATA Enter Data State

AT+CGDATA	Enter Data State
Command Function	The execution command causes the MT to perform whatever actions are necessary to establish communication between the TE and the network using one or more GPRS PDP types.
Syntax Query	AT+CGDATA=?
Syntax	+CGDATA: “PPP” OK
Write Syntax	AT+CGDATA=<L2P>,<cid>
Write Response	CONNECT
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<L2P>	“PPP”
<cid>	numeric value of PDP context activation
Notes	Supported value for <L2P>: “PPP”.

AT+CGDCONT Define PDP Context

AT+CGDCONT	Define PDP Context
Command Function	Specifies PDP context parameter values for a PDP context identified by the (local) context identification parameter, <cid>.
Syntax Query	AT+CGDCONT=?
Syntax	+CGDCONT: (1-6),"IP",,,(0),(0,1) OK
Write Syntax	AT+CGDCONT=<cid>,<PDP_ Type>,<APN>,<PDP_ADDR>,<d_comp>,<h_ comp>
Write Response	OK
Read Syntax	AT+CGDCONT?
Read Response	+CGDCONT: <cid>,<PDP Type>,<”APN”>,<”PDP_ADDR”>,<d_comp>,<h_ comp> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<cid>	PDP Context Identifier a numeric parameter which specifies a particular PDP context definition. The parameter is local to the TE-MT interface and is used in other PDP context-related commands. The range of permitted values (minimum value = 1) is returned by the test form of the command.
<PDP_type>	“IP”
<”APN”>	“Access Point Name”
<”PDP_addr”>	” Identifies the MT in the address space”

AT+CGDCONT	Define PDP Context
<d_comp>	0 = off 1 = on
<d_comp> (if HS 1000)	A numeric parameter that controls PDP data compression (applicable for SNDCP only). Currently AUF IDL does not support PDP data compression, that?s why the supported value is 0. It is an optional parameter. AT parser will ignore this parameter 0 = off Note: Currently AUF IDL does not support PDP header compression, that?s why the supported value is 0. AT parser will ignore this parameter.
<h_comp>	0 = off 1 = on
<h_comp> (if HS 1000)	A numeric parameter that controls PDP header compression. 0 = off
Notes	AT+CGDCONT must be entered before Context activation. AT+CGDCONT=1,”IP”,””,””,0,0 may be entered for networks that dynamically assign the APN. Contact your service provider for correct APN information.

AT+CGEREP GPRS Event Reporting

AT+CGEREP	GPRS Event Reporting
Command Function	Set command enables or disables sending of unsolicited result codes, +CGEV: XXX from MT to TE in the case of certain events occurring in the GPRS MT or the network.
Syntax Query	AT+CGEREP=?
Syntax	+CGEREP: (0-2),(0,1) OK
Write Syntax	AT+CGEREP=<mode>,<bfr>
Write Response	OK
Read Syntax	AT+CGEREP?
Read Response	+CGEREP: 0,0
	OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = buffer unsolicited result codes in the MT 1 = discard unsolicited result codes when MT-TE link is reserved (e.g. in on-line data mode); otherwise forward them directly to the TE 2 = buffer unsolicited result codes in the MT when MT-TE link is reserved (e.g. in on-line data mode) and flush them to the TE when MT-TE link becomes available; otherwise forward them directly to the TE
<bfr>	0 = MT buffer of unsolicited result codes defined within this command is cleared when <mode> 1 or 2 is entered 1 = MT buffer of unsolicited result codes defined within this command is flushed to the TE when <mode> 1 or 2 is entered (OK response shall be given before flushing the codes)

AT+CGEREP	GPRS Event Reporting
Notes	If parameter <mode> is omitted it is assumed to be the value of the last command execution or the default value (0). If parameter <bfr> is omitted it is assumed to be the value of the last command execution or the default value (0).

AT+CGEV Deactivation Indication Reporting

AT+CGEV	Deactivation Indication Reporting
Command Function	When event reporting is enabled (using AT+CGEREP), in case the AT parser receives the asynchronous event Deactivate Indication initiated from the network, the AT parser will send the AT+CGEV command to the Terminal to notify this event.
Syntax Query	N/A
Response	N/A
Write Syntax	N/A
Response	N/A
Read Syntax	N/A
Response	N/A
Execute Syntax	+CGEV
Response	+CGEV: NW DEACT <PDP_type>,<PDP_addr>,[<cid>]
Parameter Values	N/A
Notes	

AT+CGPADDR Show PDP Address

AT+CGPADDR	Show PDP Address
Command Function	The execution command returns a list of PDP addresses for the specified context identifiers.
Syntax Query	AT+CGPADDR=?
Syntax	+CGPADDR: (1) OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CGPADDR=<cid>
Execute Response	+CGPADDR: 1 OK
Unsolicited Response	N/A
Parameter Values	
<cid>	numeric value of PDP context activation
Notes	N/A

AT+CGQMIN Quality of Service Profile (Minimum Acceptable)

AT+CGQMIN	Quality of Service Profile (Minimum Acceptable)
Command Function	Allows the TE to specify a minimum acceptable profile which is checked by the MT against the negotiated profile returned in the Activate PDP Context Accept message.
Syntax Query	AT+CGQMIN=?
Syntax	+CGQMIN: "IP",(1-3),(1-4),(1-5),(1-9),(1-18,31) OK
Write Syntax	AT=CGQMIN=<cid>,<precedence>,<delay>,<reliability>,<peak>,<mean>
Write Response	OK
Read Syntax	AT+CGQMIN?
Read Response	+CGQMIN: 1,0,0,0,0,0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<cid>	> numeric value of PDP context activation
<precedence class>	1-3
<delay class>	1-4
<reliability class>	1-5
<peak throughput>	1-9
<mean throughput>	1-18,31
Notes	For any parameter where network subscribed is desired, enter 0. A special form of the set command, +CGQREQ=,… or +CGQMIN=,… provide a set of the default values of Quality of Service Profile for new PDP context definitions. AT+CGDCONT must be entered prior to entering AT+CGQMIN command.

AT+CGQREQ Quality of Service Profile (Requested)

AT+CGQREQ	Quality of Service Profile (Requested)
Command Function	Allows the TE to specify a Quality of Service Profile that is used when the MT sends an Activate PDP Context Request message to the network.
Syntax Query	AT+CGQREQ=?
Syntax	+CGQREQ: "IP",(1-3),(1-4),(1-5),(1-9),(1-18,31) OK
Write Syntax	AT+CGQREQ=<cid>,<precedence>,<delay>,<reliability.>,<peak>,<mean>
Write Response	OK
Read Syntax	AT+CGQREQ?
Read Response	+CGQREQ: 1,0,0,0,0,0 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<cid>	numeric value of PDP context activation
<precedence class>	1-3
<delay class>	1-4
<reliability class>	1-5
<peak throughput>	1-9
<mean throughput>	1-18,31
Notes	For any parameter where network subscribed is desired, enter 0. A special form of the set command, +CGQREQ=,… or +CGQMIN=,… provide a set of the default values of Quality of Service Profile for new PDP context definitions. AT+CGDCONT must be entered into the modem prior to entering AT+CGQREQ command.

AT+CGSMS Select Service for MO SMS Messages

AT+CGSMS	Select Service for MO SMS Messages
Command Function	The set command is used to specify the service or service preference that the MT will use to send MO SMS messages.
Syntax Query	AT+CGSMS=?
Syntax	+CGSMS: (0-3) OK
Write Syntax	AT+CGSMS=<service>
Write Response	OK
Read Syntax	AT+CGSMS?
Read Response	:+CGSMS: 3 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<service>	0 = GPRS 1 = circuit switched 2 = GPRS preferred (use circuit switched if GPRS not available) 3 = circuit switched preferred (use GPRS if circuit switched not available)
Notes	If parameter <service> is omitted the command does nothing. SMS over GPRS has not been fully tested.

ATD Request GPRS Service

ATD	Request GPRS Service
Command Function	This command causes the MT to perform whatever actions are necessary to establish communication between the TE and the external PDN
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	ATD<GPRS_SC><CID>#
Execute Response	CONNECT
Unsolicited Response	N/A
Parameter Values	
<GPRS_SC>	*99
<CID>	***1 ***2
Notes	ATD*99***1# - Dials GPRS call for context activation 1. See +CGDCONT for context activation definition.

Audio Commands

Basic Audio Commands

AT$VEAR EAR Gain

AT$VEAR	EAR Gain
Command Function	This command is used to set the EAR gain
Syntax Query	AT$VEAR=?
Syntax	$VEAR: (0-1) OK
Write Syntax	AT$VEAR=<gain>
Write Response	OK
Read Syntax	AT$VEAR?
Read Response	$VEAR: <gain> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<gain>	0 = -11 dB 1 = 1 dB
Notes	Only effective for $VSELECT values of 0 or 3

AT$VGR Radio Modem-Microphone Receiver Gain

AT$VGR	Microphone Receiver Gain
Command Function	This command sets the receive level gain for the microphone input.
Syntax Query	AT$VGR=?
Syntax	$VGR: (0-24) OK
Write Syntax	AT$VGR=<rxgain>
Write Response	$VGR: <rxgain> OK
Read Syntax	AT$VGR?
Read Response	$VGR: <rxgain> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<rxgain>	0-12 dB 1-11 dB 2-10 dB … … 24+12 dB
Notes	Receiver gain settings are in 1 dB steps from –12 to +12 dB.

AT$VGT Speaker Transmit Gain

AT$VGT	Speaker Transmit Gain
Command Function	This command is used to set the coarse speaker transmit gain
Syntax Query	AT$VGT=?
Syntax	$VGT: (0-12) OK
Write Syntax	AT$VGT=<txgain>
Write Response	$VGT: <txgain> OK
Read Syntax	AT$VGT?
Read Response	$VGT: <txgain>
	OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<txgain>	0 = -6 dB 1 = -5 dB 2 = -4 dB 3 = -3 dB … - … 12 = +6 dB
Notes	Tx gain settings in 1 dB steps from –6 to +6 dB.

AT$VLVL Speaker Volume

AT$VLVL	Speaker Volume
Command Function	This command is used to set the speaker volume
Syntax Query	AT$VLVL=?
Syntax	$VLVL: (0-5) OK
Write Syntax	AT$VLVL=<volume>
Write Response	OK
Read Syntax	AT$VLVL?
Read Response	$VLVL: <volume> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<volume>	0 = Mute 1 = -24 dB 2 = -18 dB 3 = -12 dB 4 = -6 dB 5 = 0 dB
Notes	N/A

AT$VST Sidetone Volume

AT$VST	Sidetone Volume
Command Function	This command is used to set the sidetone volume
Syntax Query	AT$VST=?
Syntax	$VST: (0-10) OK
Write Syntax	AT$VST=<sidetone level>
Write Response	OK
Read Syntax	AT$VST
Read Response	$VST: =<sidetone level> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<sidetone level>	0 = Mute 1 = -23 2 = -20 dB 3 = -17 dB 4 = -14 dB 5 = -11 dB 6 = -8 dB 7 = -5 dB 8 = -2 dB 9 = +1 Db
Notes	N/A

Advanced Audio Commands

AT$DFIR Configure Downlink FIR Coefficients

AT$DFIR	Configure Downlink FIR Coefficients
Command Function	This command allows the user to set the downlink FIR filter coefficients to improve voice quality.
Syntax Query	AT$DFIR=?
Syntax	$DFIR: (0-FFFF),(0-FFFF),(0-FFFF) (32 entries) OK
Write Syntax	AT$DFIR =<coeff1>,<coeff2>, …<coeff31>,<coeff32>
Write Response	OK
Read Syntax	AT$DFIR?
Read Response	$DFIR: <coeff1>,<coeff2>, … (12) <coeff13>,<coeff14>, … (12) <coeff25>,<coeff26>, … (8)
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
< coeff1 >	0-ffff= 2.14 fixed point number.
< coeff2 >	0-ffff= 2.14 fixed point number.
…	…
< coeff31 >	0-ffff= 2.14 fixed point number.
< coeff32 >	0-ffff= 2.14 fixed point number.

AT$DFIR	Configure Downlink FIR Coefficients
Notes	Only 31 coefficients are required for the hardware but programs being used to generate the coefficients output 32. The fewer modifications needed to the output the better. These coefficients are 2.14 fixed point values input in hexadecimal.
Examples	AT$DFIR =4000,0,0,…,0,0 4000 followed by all zeros is unity (pass through mode).

AT$MICAEC Echo Cancellation

AT$MICAEC	Echo Cancellation
Command Function	This command allows the user to configure the echo cancellation settings for the current voice mode (see $vselect)
Syntax Query	AT$MICAEC=?
Syntax	$MICAEC: (0-2), (3,7), (0,1),(0,3), (0-3), (0-24) , (3275-32767) OK
Write Syntax	AT$MICAEC=<control>,<mode >,<cont filter>, <uplink scaling>,<downlink scaling >,<max level>,<smoothing coef>
Write Response	OK
Read Syntax	AT$MICAEC?
Read Response	$MICAEC: <control>,<mode >,<cont filter>,<uplink scaling>,<downlink scaling>,<max level>,<smoothing coef > OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
< control >	0 = stop 1 = start 2 = update
< mode >	3 = divergence control enable 7 = divergence control disable
< cont filter >	0 = disable 1 = enable
< uplink scaling >	0 = disable 3 = enable

AT$MICAEC	Echo Cancellation
< downlink scaling >	0 = enable 3 = disable
< max level >	0 = -24 dB … 24 = 0 dB
< smoothing coef >	3275 = ???? … 32767 = ????
Notes	N/A
Examples	AT$MICAEC=1,3,1,3,0,12,3275 Start AEC with divergence control enabled, continuous filtering, uplink scaling, no downlink scaling, max level = -12 dB, smoothing coef = 3275.

AT$MICANR Ambient Noise Reduction Control

AT$MICANR	Ambient Noise Reduction Control
Command Function	This command allows the user to configure the ambient noise reduction settings for the current voice mode (see $vselect)
Syntax Query	AT$MICANR=?
Syntax	$MICANR: (0-2),(0-1,3),(0-2),(0-20),(0-2) OK
Write Syntax	AT$MICANR=<control>, <mode>,<noise level>,<tone thresh>,<tone count>
Write Response	OK
Read Syntax	AT$MICANR?
Read Response	$MICANR: <control>,<mode>,<noise level>,<tone thresh>,<tone count> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
< control >	0 = stop. 1 = start. 2 = update.
< mode >	0 = ANR disabled. 1 = ANR and tone detector enabled. 3 = ANR enabled, tone detect disabled
< noise level >	0 = noise attenuation based on incoming SNR 1 = 6 dB noise attenuation 2 = 12 dB noise attenuation
< tone thresh >	0-20 7 = 21dB (recommended).

AT$MICANR	Ambient Noise Reduction Control
< tone count >	0 = no tone detection 1 = single tone detection 2 = dual tone detection (DTMF)
Notes	
Examples	AT$MICANR=1,1,1,7,1 Start ANR, ANR+Tone enabled, 6 dB attenuation, threshold 21 dB, single tone detection.

AT$MICBIAS Mic Bias

AT$MICBIAS	Mic Bias
Command Function	Enable/disable the voice uplink and/or downlink. This controls enabling the mic bias.
Syntax Query	AT$MICBIAS=?
Syntax	$MICBIAS: (0-2),(0-2) OK
Write Syntax	AT$MICBIAS=<vulon>,<vdlon>
Write Response	OK
Read Syntax	AT$MICBIAS?
Read Response	$MICBIAS:<vulon>,<vdlon> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<vulon >	0 = voice uplink off 1 = voice uplink on 2 = voice uplink no change
<vdlon >	0 = voice downlink off 1 = voice downlink on 2 = voice downlink no change
Notes	The parameters for this command are not saved by AT&W. The settings controlled by this command are not affected by AT&F.
Examples	

AT$MICES Echo Suppression Control

AT$MICES	Echo Suppression Control
Command Function	This command allows the user to enable the echo suppressor and select the behavior for the current voice mode (see AT$VSELECT)
Syntax Query	AT$MICES=?
Syntax	$MICES: (0-1), (0-6) OK
Write Syntax	AT$MICES=<control>,<behavior>
Write Response	OK
Read Syntax	AT$MICES?
Read Response	$MICES: <control>,< behavior> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<control>	0 = disable echo suppression. 1= enable echo suppression.
<behavior>	0 = Behavior 1 1 = Behavior 1a 2 = Behavior 2a 3 = Behavior 2b 4 = Behavior 2c 5 = Behavior 2c_idle 6 = Behavior 3 255 = Custom

AT$MICES	Echo Suppression Control
Notes	When <behavior> 255 (custom) is selected, the $MICESC command is used to configure the Echo Suppression parameters. Behaviors are defined in ITU-T P.340.
Examples	AT$MICES=1,3 Enable echo suppression configured with predefined behavior 2b parameters.

AT$MICESC Echo Suppressor Configuration

AT$MICESC	Echo Suppressor Configuration
Command Function	This command allows the user to set values for the echo suppressor. The echo suppressor (ES) role is to control the residual echo in a speakerphone application, where the AEC is unable to cancel the entire echo in the uplink due to non-ideal acoustical environment (such as a non-linear loudspeaker).
Syntax Query	AT$MICESC=?
Syntax	$MICESC: (0-3F),(0-FFFF) (20 entries),(0-2)
Write Syntax	AT$MICESC=<es_mode>,<es_gain_dl>,<es_gain_ul_1>,<es_gain_2>,<tcl_fd_ls_thr>,<tcl_dt_ls_thr>,<tcl_fe_ns_thr>,<tcl_fe_ns_thr>,<tcl_ne_thr>,<ref_ls_pwr>,<switching_time>,<switching_time_dt>,<hang_time>, <gain_lin_dl_vect[0]>, >,<gain_lin_dl_vect[1]>,>,<gain_lin_dl_vect[2]>, >,<gain_lin_dl_vect[3]>,>,<gain_lin_ul_vect[0]>, >,<gain_lin_ul_vect[1]>,>,<gain_lin_ul_vect[2]>, >,<gain_lin_ul_vect[3]>, [<voice_mode>]
Write Response	OK
Read Syntax	AT$MICESC?
Read Response	$MICESC: <es_mode>,<es_gain_dl>,<es_gain_ul_1>, <es_gain_2>,<tcl_fd_ls_thr>,<tcl_dt_ls_thr>,<tcl_fe_ns_thr>,<tcl_fe_ns_thr>,<tcl_ne_thr>, <ref_ls_pwr>,<switching_time>,<switching_time_dt>,<hang_time>,<gain_lin_dl_vect[0]>, >,<gain_lin_dl_vect[1]>,>,<gain_lin_dl_vect[2]>, >,<gain_lin_dl_vect[3]>,>,<gain_lin_ul_vect[0]>, >,<gain_lin_ul_vect[1]>,>,<gain_lin_ul_vect[2]>, >,<gain_lin_ul_vect[3]>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	

<table>
<tr><th>AT$MICESC</th><th>Echo Suppressor Configuration</th></tr>
<tr><td><es_mode></td><td>The following bitmap table defines the es mode:
<table>
<tr><td>BIT</td><td>Description</td></tr>
<tr><td>0 (ES UL)</td><td>0 = Disable ES on UL path
1= Enable ES on UL path</td></tr>
<tr><td>1 (ES DL)</td><td>0 = Disable ES on DL path
1 = Enable ES on DL path</td></tr>
<tr><td>2 (CNG)</td><td>0 = Disable CNG* algorithm
1 = Enable CNG* algorithm</td></tr>
<tr><td>3 (NSF)</td><td>0 = Disable NSF** algorithm
1 = Enable NSF** algorithm</td></tr>
<tr><td>4 (ALS UL)</td><td>0 = Disable ALS*** on UL path
1 = Enable ALS*** on UL path</td></tr>
<tr><td>5 (ALS DL)</td><td>0 = Disable ALS*** on DL path
1 = Enable ALS*** on DL path</td></tr>
</table>
* CNG = Comfort Noise Generation
** NSF=Noise Floor
*** ALS = Attenuation Level Smoothing

Notes:
- Disabling ES UL has no sense
- CNG and NSF must not be enabled together</td></tr>
<tr><td><es_gain_dl></td><td>Receive loss compensation.</td></tr>
<tr><td><es_gain_ul_1></td><td>Coupling loss compensation.</td></tr>
<tr><td><es_gain_ul_2></td><td>Near-end propagation loss compensation.</td></tr>
<tr><td><tcl_fe_ls_thr></td><td>TCL reference threshold in far-end mode for loud signals.
This value is in Q15 format.</td></tr>
<tr><td><tcl_dt_ls_thr></td><td>TCL reference threshold in double-talk mode for loud signals.
This value is in Q15 format</td></tr>
</table>

AT$MICESC	Echo Suppressor Configuration
<tcl_fe_ns_thr>	TCL reference threshold in far-end mode for nominal signals. This value is in Q15 format
<tcl_dt_ns_thr>	TCL reference threshold in double-talk mode for nominal signals. This value is in Q15 format
<tcl_ne_thr>	TCL reference threshold in near-end mode. This value is in Q15 format.
<ref_ls_pwr>	TCL reference threshold in near-end mode. This value is in Q15 format
<switching_time>	The switching time value in milliseconds.
<switching_time_dt>	The double-talk switching time value in milliseconds.
<hang_time>	The hangover time for switching.
<gain_lin_dl_vect[0-3]>	Table containing downlink linear attenuation levels per state: gain_lin_dl_vect[0] - idle state gain_lin_dl_vect[1] - double talk gain_lin_dl_vect[2] - far-end gain_lin_dl_vect[3] - near-end Format is Q15.
<gain_lin_ul_vect[0-3]>	Table containing uplink linear attenuation levels per state: gain_lin_ul_vect[0] - idle state gain_lin_ul_vect[1] - double talk gain_lin_ul_vect[2] - far-end gain_lin_ul_vect[3] - near-end Format is Q15.
<voice_mode>	0 = Set parameters for handset voice mode 1 = Set parameters for headset voice mode Default is to set the parameters for the current voice mode.
Notes	N/A

AT$PREAMP Set Uplink Voice Parameters

AT$PREAMP	Set Uplink Voice Parameters
Command Function	This command allows the user to enter uplink voice specific parameters for the current voice mode (see $vselect).
Syntax Query	AT$PREAMP=?
Syntax	$PREAMP: (0-1),(0-24),(0,8) OK
Write Syntax	AT+PREAMP=<bias>, <gain>,<extra gain>
Write Response	OK
Read Syntax	AT$PREAMP?
Read Response	$PREAMP: <bias>,<gain>,<extra gain>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<bias>	0 = 2v. 1 = 2.5v.
<gain>	The value of the gain follows: 0 = -12 dB 1 = -11 dB 2 = -10 dB 3 = -9 dB . . . 21 = 9 dB 22 = 10 dB 23 = 11 dB 24 = 12 dB

AT$PREAMP	Set Uplink Voice Parameters
<extra gain>	0 = -2 dB. 1 = 0 dB. 2 = 2 dB. 3 = 4 dB. 4 = 6 dB. 5 = 8 dB. 6 = 10 dB. 7 = 12 dB. 8 = 14 dB.
Notes	Change in bias may or may not have an Effect, depending on hardware. Extra gain is not supported in headset mode. Changing the value in headset mode will have no affect on the module configuration.
Examples	AT$PREAMP =1,24,8 Max volume from the microphone.

AT$SPKCFG Set Downlink Voice Parameters

AT$SPKCFG	Set Downlink Voice Parameters
Command Function	This command allows the user to configure the downlink voice path parameters for the current voice mode (see $vselect).
Syntax Query	AT$SPKCFG=?
Syntax	$SPKCFG: (0-12),(0-5),(0-1),(0-1),(0-1),(0-1,3) OK
Write Syntax	AT$SPKCFG=<gain>,<volume>,<filter>,<highpass filter>,<headset Ov Mid>,<voice mode>
Write Response	OK
	Note: The current <headsetOvMid> and current <voice mode> settings are assumed if not entered when writing this command.
Read Syntax	AT$SPKCFG?
Read Response	$SPKCFG: <gain>,<volume>,<filter>,<highpass filter>,<headset Ov Mid>,<voice mode>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	

AT$SPKCFG	Set Downlink Voice Parameters
< gain >	0 = -6 dB. 1 = -5 db. 2 = -4 db. 3 = -3 db. 4 = -2 db. 5 = -1 db. 6 = 0 db. 7 = 0 db. 8 = 2 db. 9 = 3 db. 10 = 3 db. 11 = 5 db. 12 = 6 db.
< volume >	The value of volume is as follows: 0 = Mute 1 = -24 dB 2 = -18 dB 3 = -12 dB 4 = -6 db 5 = 0 dB
< filter >	0 - on 1 - off Enable/disable voice filter. Filter coefficients set by $DFIR/$UFIR commands
<highpass filter >	0 – on 1 – off

AT$SPKCFG	Set Downlink Voice Parameters
<headset Ov Mid>	0 - no pseudo ground is provided on terminal HSOVMID (Pin 61) 1 - a pseudo ground is provided on terminal HSOVMID (Pin 61) Note: Please refer to the Integration Guide for proper use of the HSOVMID (headset output common mode feature.)
<voice mode>	0 - Selects handset for voice 1 - Selects headset for voice

AT$UFIR Configure Uplink FIR Coefficients

AT$UFIR	Configure Uplink FIR Coefficients
Command Function	This command allows the user to set the uplink FIR filter coefficients to improve voice quality.
Syntax Query	AT$UFIR=?
Syntax	$UFIR: (0-FFFF),(0-FFFF),(0-FFFF) (32 entries) OK
Write Syntax	AT$UFIR =<coeff1>,<coeff2>, …<coeff31>,<coeff32>
Write Response	OK
Read Syntax	AT$UFIR?
Read Response	$UFIR: <coeff1>,<coeff2>, … (12) <coeff13>,<coeff14>, … (12) <coeff25>,<coeff26>, … (8)
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
< coeff1 >	0-ffff=> 2.14 fixed point number.
< coeff2 >	0-ffff=> 2.14 fixed point number.
…	
< coeff31 >	0-ffff=> 2.14 fixed point number.
< coeff32 >	0-ffff=> 2.14 fixed point number.
Notes	Only 31 coefficients are required for the hw but programs being used to generate the coefficients output 32. The fewer modifications needed to the output the better. These coefficients are 2.14 fixed point values input in hexadecimal. Examples AT$UFIR =4000,0,0,…,0,0 4000 followed by all zeros is unity (pass through mode).

AT$VSELECT Voice Select

AT$VSELECT	Voice Select
Command Function	This command selects the voice mode of the device. Only valid options applicable to the hardware will be allowed. All applicable constants and settings are loaded when the mode is changed and at power up.
Syntax Query	AT$VSELECT=?
Syntax	$VSELECT: (0,1,3) OK
Write Syntax	AT$VSELECT= <mode>
Write Response	OK
Read Syntax	AT$VSELECT?
Read Response	$VSELECT: 0
Execute Syntax	AT$VSELECT
Execute Response	$VSELECT : <reset state> OK
Unsolicited Response	N/A
Parameter Values	
<Mode>	0 = Selects handset for voice 1 = Selects headset for voice 3 = Automatic
Notes	In Automatic mode ($VSELECT=3), the device will default to handset mode. If a headset is plugged into the headset jack, the device will automatically switch to headset mode. When the headset is removed, the device will switch back to handset mode. AT$VSELECT=3 must be entered prior to plugging in the headset, to switch between modes properly.
Examples	To set the voice mode to Headset: AT$VSELECT=1 OK

GPIO Commands

AT$IGNDBNC Ignition Debounce

AT$IGNDBNC	Ignition Debounce
Command Function	This command allows the user to set ignition debounce time used for the event engine. The ignition line has to be valid for the specified amount of time before the event: GPIO-8 in the event engine will be triggered.
Syntax Query	AT$IGNDBNC=?
Syntax	$IGNDBNC: (1-4) OK
Write Syntax	AT$IGNDBNC=<debounceTimeout>
Write Response	OK
Read Syntax	AT$IGNDBNC?
Read Response	$IGNDBNC: 0
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<delay>	1 – 4 seconds. This field specifies the debounce timeout value.
Notes:	The "reset upon ignition" reset interval will follow the setting of AT$IGNDBNC. $IODBNC is affected by this command. Changes made affect GPIO8 in $IGNDBNC? writes made by $IGNDBNC will be seen by $IODBNC?

AT$IOCFG GPIO Configuration

AT$IOCFG	GPIO Configuration
Command Function	This command is used to query the GPIO direction setting (input or output).
Syntax Query	AT$IOCFG=?
Syntax	$IOCFG: (000000000-111111111) OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	AT$IOCFG?
Read Response	$IOCFG: <current setting> <configured setting> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = OUTPUT 1 = INPUT

AT$IOCFG	GPIO Configuration
Notes	9 Digits where left bit represents GPIO 1 and right most bit represents GPIO 9. GPIO 8 should NOT be set to Output because it is the ignition sense. GPIO 5 should NOT be set to Input because it is used by the GPS receiver. GPIO 3 - 9 cannot be changed. 1 = GPI1 input only 2 = GPO2 output only 3 = GPO3 output only, latched 4 = LED for GPS user 3 LED (controlled by MSP, set when $GPSCMD is set) 5 = GPO5 output only 6 = USR1 LED Registration with default events 7 = USR2 LED GPS lock with default events 8 = ignition sense input only 9 = GPI9 (when selected with AT$RPTADC) This is always GPI9 AND ADC2, even when $RPTADC=1. The $RPTADC setting only controls whether the output msg contains the ADC value or a digital value based on reading ADC2. Note: Non latched IOs are undefined during reset.
Example:	Output format for read command AT$IOCFG? $IOCFG: 11110111 10110111 OK Configured I/O settings where I/O pins 1,3,4,6,7, and 8 are inputs and pins 2 and 5 are outputs. Current I/O settings where I/O pin 2 has been changed to an input line.

AT$IOADC# Read Analog to Digital Converter

AT$IOADC#	Read Analog to Digital Converter
Command Function	This command returns the raw, uncalibrated, ADC value of the ADCIN signal. The number of supported IOs varies by device. The command format is AT$IOADC# (where # represents the specified I/O - When using this command, replace "#" with the actual number.)
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$IOADC#
Execute Response	$IOADC#: <value> OK
Unsolicited Response	N/A
Parameter Values	
<value>	Raw ADC value in decimal format. Range is 0 to 1023.
Notes	See Enabler III Integration Guide for pinout definitions. If the modem is not registered or attached, the modem may take up to a minute to update the digital output to reflect a change on the analog input. Range: 0-16V Resolution 10 bit 0.0156 V per bit Resistor divider is +/-1%. +/-2 LSB therefore if precision is needed, Enfora recommenda that you calibrate to a known voltage. There will still be some drift with temperature and there may be drift over time so recalibration is also recommended in applications that require precision.

AT$IODBNC Debounce specified GPIO for the specified amount of time

AT$IODBNC	Debounce specified GPIO for the specified amount of time
Command Function	This command allows a user to set and query GPIO debounce time. The GPIO must be unchanged for the specified number of seconds before the input event will be triggered.
Syntax Query	AT$IODBNC=?
Syntax	$IODBNC: (1-9),(0-60) OK
Write Syntax	AT$IODBNC=<gpio_number>,<debounce_timeout>
Write Response	OK
Read Syntax	AT$IODBNC?
Read Response	$IODBNC: <gpio1>,<gpio2>,<gpio3>,<gpio4>,<gpio5>,<gpio6>,<gpio7>,<gpio8>, <gpio9> OK
Execute Syntax	AT$IODBNC
Execute Response	ERROR
Unsolicited Response	N/A
Parameter Values	
<gpio_number>	Number of GPIO whose debounce timeout is being set.
<debounce_timeout>	Number of consecutive seconds <gpio_number> must remain unchanged before its input event will be triggered.
<gpio1>	Debounce timeout for GPIO1.
<gpio2>	Debounce timeout for GPIO2.
<gpio3>	Debounce timeout for GPIO3.
<gpio4>	Debounce timeout for GPIO4.

AT$IODBNC	Debounce specified GPIO for the specified amount of time
<gpio5>	Debounce timeout for GPIO5.
<gpio6>	Debounce timeout for GPIO6.
<gpio7>	Debounce timeout for GPIO7.
<gpio8>	Debounce timeout for GPIO8.
<gpio9>	Debounce timeout for GPIO9.
Notes	If <debounce_timeout> is set to zero, <gpio_number> will not be debounced. $IGNDBNC is affected by this command. Changes made to GPIO8 will be seen via $IGNDBNC? Writes made by $IGNDBNC will be seen by $IODBNC? Regardless of the AT$IGNDBNC setting or the AT$IODBNC=8, <debounce_ timeout> setting, if the ignition signal has gone from logic zero to a logic one for longer than 1 second, the unit will be reset 3 seconds later. AT$IODBNC=8,<debounce_timeout> should not be set for any time greater than 1.

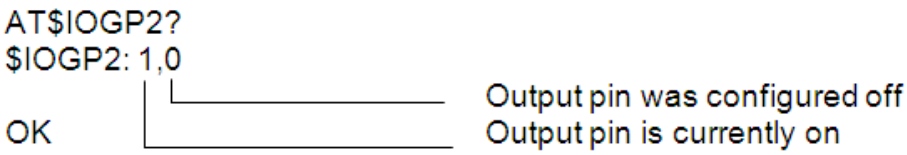
AT$IOGPA GPIO Byte Control

AT$IOGPA	GPIO Byte Control
Command Function	This command allows the user to set the state of all GPIO bits simultaneously. Only GPIO pins previously configured as outputs will be effected. (See AT$IOCFG)
Syntax Query	AT$IOGPA=?
Syntax	$IOGPA: (000000000-111111111) OK
Write Syntax	AT$IOGPA=<mode>
Write Response	OK
Read Syntax	AT$IOGPA?

AT$IOGPA	GPIO Byte Control
Read Response	$IOGPA: <current setting> <configured setting> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = off 1 = on
Notes	9 Digits where left bit represents GPIO 1 and right most bit represents GPIO 9. Do NOT modify GPIO5 (see IOGP(x) note).
Example	Output format for read command AT$IOGPA? $IOGPA: 11111111 00000000 OK Configured I/O settings where I/O pins are all off. Current I/O settings where I/O pins are all on.

AT$IOGP(x) GPIO Bit Control

AT$IOGP(x)	GPIO Bit Control
Command Function	This command allows the user to set the state of the specified GPIO bit. The GPIO being written to must have previously been set to an output. (See AT$IOCFG).
Syntax Query	AT$IOGP(x)=?
Syntax	$IOGP(x): (0-1) OK
Write Syntax	AT$IOGP(x)=<mode>
Write Response	OK
Read Syntax	AT$IOGP(x)?
Read Response	$IOGP(x): <current setting> <configured setting> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
(x)	1-8 GPIO bit
<mode>	0 = off 1 = on
Notes	GPIO5 is reserved. It is a general-purpose output, and controls the RESET line of the GPS receiver. Set HIGH for normal operation, LOW to RESET the GPS receiver used to control the GPS module. AT$IOGP5=? returns ERROR AT$IOGP5? returns $GPS_Status:1 for GPS on or 0 for GPS off. see AT$GPSCMD? For possible response values.

AT$IOGP(x)	GPIO Bit Control
Example:	Output format for read command 

AT$IOPULUP GPIO Pull-up Settings

AT$IOPULUP	GPIO Pull-up Settings
Command Function	This command allows the user to set the GPIO pullup state on GPIO 1 and GPIO 9.
Syntax Query	AT$IOPULUP=?
Syntax	$IOPULUP: (0-1),(0-1) OK
Write Syntax	AT$IOPULUP =<pullup_state_GPIO1><pullup_state_GPIO9>
Write Response	OK
Read Syntax	AT$IOPULUP?
Read Response	$IOPULUP: < pullup_state_GPIO1><pullup_state_GPIO9> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<pullup_state_GPI1>	0 = pulldown. 1 = pullup.
<pullup_state_GPI9>	0 = pulldown. 1 = pullup.
Notes	N/A

AT$IOTAR Input/Output Transition Across Reset

AT$IOTAR	Input/Output Transition Across Reset
Command Function	Ability to detect GPIO transition that occurs during a modem reset This command allows the user to set/query the flags that control how the event engine's input GPIO events are initialized. By default, the modem initializes the GPIO input events to the value detected in the hardware when the code is initialized. So the first pass through the event engine logic generally does not trigger a transition event (see $EVENT). Setting the appropriate flag with this command allows the code to initialize the GPIO input event with the last value saved to flash prior to the modem reset. Using this method allows the modem to generate a transition event, if the GPIO has changed state while the modem was resetting.
Syntax Query	AT$IOTAR=?
Syntax	$IOTAR: (000000000-111111111) OK
Write Syntax	AT$IOTAR=<iotar_config>
Write Response	OK
Read Syntax	AT$IOTAR?
Read Response	$IOTAR: <iotar_config>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<iotar_config>	9 digits where left most bit represents GPIO 1 and right most bit represents GPIO 9. Where a digit of 0 (default) results in the associated input event getting initialized with the current hardware value, and a value of 1 will initialize the input event using the value saved in FFS prior to reset.

AT$IOTAR	Input/Output Transition Across Reset
Notes	N/A
Example	AT$IOTAR=000000001 OK AT$IOTAR? $IOTAR: 000000001 OK

AT$RPTADC Report ADC2 or GPIO9

AT$RPTADC	Report ADC2 or GPIO9
Command Function	This command allows the user to select the data that will be sent over the air when bit 5 of parm2 in the Bit-Field Table is selected
Syntax Query	AT$RPTADC=?
Syntax	$RPTADC:(0-1) OK
Write Syntax	AT$RPTADC=<reported value>
Write Response	OK
Read Syntax	AT$RPTADC?
Read Response	$RPTADC:<reported value>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<reported value>	Value to report over the air. 0 – GPIO9 1 – ADC2
Notes	N/A.

IP Router Commands

AT$GATEWAY Gateway IP

AT$GATEWAY	Gateway IP
Command Function	This command allows the user to select a gateway IP. Windows CE 3.0 devices and some Linux platforms require a gateway address. Default value "0.0.0.0" indicates that no gateway IP will be requested from the host. A non-zero value will cause the modem to request the indicated gateway IP from the host.
Syntax Query	AT$GATEWAY=?
Syntax	$GATEWAY: ("<IP>") OK
Write Syntax	AT$GATEWAY ="<IP >"
Write Response	OK
Read Syntax	AT$GATEWAY?
Read Response	$GATEWAY: "<IP >"
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<IP>	gateway IP address.
Notes	N/A

AT$DISC Disconnect Network Connection

AT$DISC	Disconnect Network Connection
Command Function	This command allows the user to initiate a network disconnect. This command is only valid for AT$HOSTIF=3 after the local PPP connection has been established or over-the-air as an API command when in TCP PAD mode.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$DISC
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	This command will only disconnect the network connection when AT$HOSTIF=3. The local PPP connection will remain active. This command can also be used to function as a disconnect request for TCP PAD. It must be sent over the air using the UDPAPI AT Command write sequence

AT$HOSTIF Configure Host to Modem Interface

AT$HOSTIF	Configure Host to Modem Interface
Command Function	This command allows the user to configure the desired Host to Modem interface. This parameter determines the behavior of the ATD command.
Syntax Query	AT$HOSTIF=?
Syntax	(0-3)
Write Syntax	AT$HOSTIF=<host interface>
Write Response	OK
Read Syntax	AT$HOSTIF=?
Read Response	HOSTIF: <host interface>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<host interface>	0 = Establish normal external Dial up networking modem to network connection. 1 = Establish UDP PAD session. Upon establishment of a network activation, a CONNECT message will be displayed. “No Carrier” or error will indicate failed or terminated UDP PAD session. 2 = Establish TCP PAD session Upon establishment of a network activation, a CONNECT message for at$active=1, or a LISTEN message for at$active=0 will be displayed. “No Carrier” or error will indicate failed or terminated TCP PAD session. 3 = Establish non-GPRS PPP connection. (Not supported on USB)

AT$HOSTIF	Configure Host to Modem Interface
Notes	When HOSTIF = 3, all port connection requests must originate from the Host system. When the modem is configured for this mode, it is operating as a non-configurable router / firewall. FTP active mode is not supported. Some programs may require a remote proxy in order to work. Do not use AT$AREG=2 with autobauding of the serial port and TCP or UDP PAD functions. The serial port will not respond to at commands if the modem establishes a connect state before the baud rate has been determined for the serial port.

AT$NETIP Display Network Assigned IP & DNS

AT$NETIP	Display Network Assigned IP & DNS
Command Function	This command allows the user to query the modem's network assigned IP.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	AT$NETIP?
Read Response	<"IP">,<"DNS1">,<"DNS2">
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<IP>	network assigned IP
<DNS1>	network assigned DNS1
<DNS2>	network assigned DNS2
Notes	N/A

AT$FLFILT - Filter ICMP Echo Request

AT$FLFILT	Filter ICMP Echo Request
Command Function	This command allows the user to send an ICMP echo request and display the echo reply (ping).
Syntax Query	AT$FLFILT=?
Syntax	$FLFILT: (0-1) OK
Write Syntax	N/A
Write Response	N/A

AT$FLFILT	Filter ICMP Echo Request
Read Syntax	AT$FLFILT?
Read Response	$FLFILT: <enable> OK
Execute Syntax	AT$FLFILT=<enable>
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	
<enable>	0 = do not filter ICMP echo requests 1 = filter ICMP echo requests against Friends List ** Default value is 1
Notes	Friends list entries are used without regard to the setting of the server bit. This function will also filter ICMP Echo requests when in Dial-Up Networking Mode ($HOSTIF=0)

AT$PING - Send ICMP Echo Request

AT$PING	Send ICMP Echo Request
Command Function	This command allows the user to send an ICMP echo request and display the echo reply (ping).
Syntax Query	AT$PING=?
Syntax	$PING: “host”,(1-5),(1-10) OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$PING=<”host”>,<count>,<timeout>

AT$PING	Send ICMP Echo Request
Execute Response	Immediate response: “Pinging <IP address>” After ICMP response: “Reply from <IP address>”
Unsolicited Response	N/A
Parameter Values	
<"host">	FQDN or IP address
<count>	Number of ICMP frames to send Range: 1-5 Default: 1
<timeout>	Maximum amount of time, in seconds, to wait for a response. Range: 1-10 Default: 5
Notes	The modem must have an IP address assigned to it by the network to send or receive ICMP packets.

Message Commands

AT$APIOPT Enable API Optional Header Fields

AT$APIOPT	Enable API Optional Header Fields
Command Function	This command allows the user to enable specific Optional Header Fields to be included in the UDPAPI and TCPAPIs' API Optional Header. See Enfora GSM/GPRS Family API Reference (GSM0308UG001) for details.
Syntax Query	AT$APIOPT=?
Syntax	$APIOPT: (0-1),(0-1),(0-4),(0-1),(0-1),(0-1) OK
Write Syntax	AT$APIOPT=<MDMID>,<Msg Event Format>,<Event Seq Num>,<HdrDisable>,<Output Event Type>,<HexModemID>
Write Response	OK
Read Syntax	AT$APIOPT?
Read Response	$APIOPT=<MDMID>,<Msg Event Format>,<Event Seq Num>,<HdrDisable>,<Output Event Type>,<HexModemID>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<MDMID>	0 = Disable sending of MDMID value in TCPAPI or UDPAPI Header 1 = Enable sending of MDMID value in TCPAPI or UDPAPI Header
<Msg Event Format>	0 = Disable sending of Output Message Event Format value in TCPAPI or UDPAPI Header 1 = Enable sending of Output Message Event Format in TCPAPI or UDPAPI Header

AT$APIOPT	Enable API Optional Header Fields
<Event Seq Num>	0 = Disable sending of Event Sequence Number value in TCPAPI or UDPAPI Header 1 = Enable sending of the least significant byte of the Event Sequence Number in TCPAPI or UDPAPI Header 2 = Enable sending of the two least significant bytes of the Event Sequence Number in TCPAPI or UDPAPI Header 3 = Enable sending of the three least significant bytes of the Event Sequence Number in TCPAPI or UDPAPI Header 4 = Enable sending of the full four bytes of the Event Sequence Number in TCPAPI or UDPAPI Header
<HdrDisable>	0 = UDPAPI header is enabled (default) 1 = UDPAPI header is disabled for UDP messages sent via event engine. The UDPAPI header will not be disabled if bit 18 (send this OTA message via SMS when GPRS services are not available) is set in parm2 of the $event command that generates the message (see AT$EVENT). This flag has no effect on commands sent from the server or the ack sequence. This feature is intended to be used with UDP messages when the size of the data packet is critical due to the rapid rate at which messages are sent over an extended period of time.
<Output Event Type>	0 = Disable sending of Output Event number in TCPAPI or UDPAPI Header 1 = Enable sending of Output Event number in TCPAPI or UDPAPI Header
<HexModemID>	0 = Do not include the hex modem ID in the optional header 1 = Include the hex modem ID in the optional header Note: The optional header type for hex modem ID is 7.
Notes	When a message is formatted for transmission via the event engine (UDP, TCP, SMS, or serial), we call the function to retrieve/increment the sequence number. The sequence number is passed to the task that actually makes the decision about what to do with the message. If the message is SMS or serial, the sequence number doesn't actually get used in the end, but does get incremented.

AT$APIOPT	Enable API Optional Header Fields
Examples	This example will disable the sending of the MDMID, enable the Msg Event Format and enable the full Event Sequence Number. AT$APIOPT=0,1,4 This example will enable the sending of the MDMID and the least two significant bytes of the Event Sequence Number. The state of the Enable Msg Event Format will remain unchanged AT$APIOPT=1,,2 This example will enable the sending of the MDMID and disable the Msg Event Format. The state of the Event Seq Num will remain unchanged AT$APIOPT=1,0 Hex Modem ID Examples: MDMID - "001036000211071" {07}{07}{F1}{36}{6C}{B0}{7F} MDMID - "99" {03}{07}{63} MDMID - "999999999999999999A" {03}{07}{00}

AT$APIPWD API Password

AT$APIPWD	API Password
Command Function	This command allows the user to query/set the API password. Any IP address will be able to access the device if the password is not set (null). If the password is set, control is given to the last friend IP that supplied the correct password.
Syntax Query	AT$APIPWD=?
Syntax	$APIPWD: ("PASSWORD") OK
Write Syntax	AT$APIPWD=”<API password>”
Write Response	OK
Read Syntax	AT$APIPWD?
Read Response	$APIPWD: "<API password>”
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<API password>	Maximum of 8 characters. A NULL password indicates ALL remote users are allowed API access.
Notes	See Enfora GSM-GPRS Family UDP-API Reference GSM0308UG001 for further details regarding the use of the API Password.

AT$ATPASSWD Set authorization for AT commands for serial, SMS and API

AT$ATPASSWD	Set authorization for AT commands for serial, SMS and API
Command Function	This command allows the user to enable or disable authorization for AT commands for the serial, SMS and API. It also sets the password required to run this command
Syntax Query	AT$ATPASSWD=?
Syntax	AT$ATPASSWD: (“oldpasswd”,mask|”newpasswd”) OK
Write Syntax	AT$ATPASSWD= <”oldpasswd”,”newpasswd”>
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax Execute Response	AT$ATPASSWD=<”passwd”,mask>
Unsolicited Response	N/A
Response	N/A
Parameter Values	
<oldpasswd>	Specified when the password is being changed.
<newpasswd	This is the value of the new password and is specified only when the password is being changed. It must be no more than eight characters in length and must be enclosed in double quotes.
<passwd>	Specified when changing the AT command authorization mask.

<table>
<tr><th>AT$ATPASSWD</th><th>Set authorization for AT commands for serial, SMS and API</th></tr>
<tr><td><mask></td><td>Bit mask specifying which interfaces will be authorized to enter AT commands. Each bit specifies one interface as enumerated in the table below.
<table><tr><td>Bit Value</td><td>Interface</td></tr><tr><td>1</td><td>Serial Port</td></tr><tr><td>2</td><td>SMS</td></tr><tr><td>4</td><td>API</td></tr></table>
To select multiple items to authorize, add the bit values of each interface to be authorized. To authorize API and SMS only, the mask value is 6 (4 + 2). AT commands entered over the serial port will not execute and will reply with ERROR.</td></tr>
<tr><td>Notes:</td><td>N/A</td></tr>
<tr><td>Example:</td><td>To set a password on a device that previously had no set $ATPASSWD enter: AT$ATPASSWD="","1234" To change a previously set $ATPASSWD password enter: AT$ATPASSWD="1234","5678"</td></tr>
</table>

AT$CHKIN Configure Device Check-in

AT$CHKIN	Configure Device Check-in
Command Function	This command adds event group 8 to the event table and configures the device check-in feature. The feature is used to send a wakeup type message to an alternate server (server other than the one the device normally reports). This alternate server is listed in the $FRIEND list with a usage=4.
Syntax Query	AT$CHKIN=?
Syntax	$CHKIN: (0-1),(0-43200) OK
Write Syntax	AT$CHKIN=<enable><periodic_interval>
Write Response	OK

AT$CHKIN	Configure Device Check-in
Read Syntax	AT$CHKIN?
Read Response	$CHKIN: 1,43200 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<enable>	0 = remove group 8 from the event table. 1 = add event group 8 to the event table.
<periodic_interval>	Sets the periodic reporting interval (in minutes) Range: 0-43200 Default: 43200
Notes	Input event 200 is used for the periodic check-in timer. Output event 152 is used for sending UDP messages to the check-in servers specified in the $FRIEND list. NOTE: Setting AT$CHKIN=1 will overwrite any existing event group 8.
Example:	AT$CHKIN=1,x adds event group 8 to the event table. AT$CHKIN=0,x removes event group 8 from the event table.

AT$FRIEND Modem Friends

AT$FRIEND	Modem Friends
Command Function	This command allows the user to configure the modem friend/server list. A friend is always allowed remote API access. Friend servers can be configured to receive WAKEUP messages whenever the modem receives a new IP, or after a certain period has elapsed. (see AT$WAKEUP)
Syntax Query	$FRIEND=?
Syntax	$FRIEND: (1-10),(0,1),"(0-255).(0-255).(0-255).(0-255)" ,(0-65535),(0-4) OK

AT$FRIEND	Modem Friends
Write Syntax	AT$FRIEND =<friend number>,<server indication>,”<friend IP> or <DNS name>”,<destination port>,<usage>
Write Response	OK
Read Syntax	AT$FRIEND?
Read Response	$FRIEND: =01, <server indication>,”<friend IP> or <DNS name>”, <destination port>, <usage> $FRIEND: =02, <server indication>,”<friend IP> or <DNS name>”,<destination port>, <usage> $FRIEND: =03, <server indication>,”<friend IP> or <DNS name>”,<destination port>, <usage> $FRIEND: =04, <server indication>,”<friend IP> or <DNS name>”,<destination port>, <usage> $FRIEND: =05, <server indication>,”<friend IP> or <DNS name>”,<destination port>, <usage> $FRIEND: =06, <server indication>,”<friend IP> or <DNS name>”,<destination port>, <usage> $FRIEND: =07, <server indication>,”<friend IP> or <DNS name>”,<destination port>, <usage> $FRIEND: =08, <server indication>,”<friend IP> or <DNS name>”,<destination port>, <usage> $FRIEND: =09, <server indication>,”<friend IP> or <DNS name>”,<destination port>, <usage> $FRIEND: =10, <server indication>,”<friend IP> or <DNS name>”,<destination port>, <usage>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<friend number>	friend identification (1-10).

AT$FRIEND	Modem Friends
<server indication>	0 = Server is disabled. 1 = Server is enabled.
<friend IP>	friend IP value.
OR	
<DNS name>	friend DNS name
<destination port>	friend destination port (TCP API only).
<usage>	0 = Unspecified (treated as UDPAPI) 1 = TCPAPI 2 = UDPAPI 3 = TCPAPI and/or UDPAPI 4 = Alternate server for UDPAPI check-in message (see AT$CHKIN)
Notes	If destination port and usage are not present in the command, it is assumed to be a UDPAPI friend with the destination port filled in with the UDPAPI port number and usage = 0. You will use either the Friend IP address or the Friend DNS name, but not both.

AT$MDMID Modem ID

AT$MDMID	Modem ID
Command Function	This command allows the user to query/set the modem ID. The modem ID is copied into each wakeup message sent from the modem. (see AT$WAKEUP)
Syntax Query	AT$MDMID=?
Syntax	$MDMID: ("MODEM ID") OK
Write Syntax	AT$MDMID =”<modem ID>”
Write Response	OK
Read Syntax	AT$MDMID?
Read Response	$MDMID: "<modem ID>”
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<modem ID >	0-20 character string in ASCII format.
Notes	The modem's IMEI is the default modem ID.

AT$MLQSIZ Message Log Queue Size

AT$MLQSIZ	Message Log Queue Size
Command Function	The $MLQSIZ allows the user to restrict the number of unsent messages maintained in the buffers for the four message log queues: UDP, UDP with SMS backup, SMS, and TCP.
Syntax Query	AT$MLQSIZ=?
Syntax	$MLQSIZ:(0-3000),(0-3000),(0-3000), (0-3000) OK
Write Syntax	AT$MLQSIZ=<UdpQsize>,<SmsBckupQsize>,<SmsQsize>,<TcpQsize>
Write Response	OK
Read Syntax	AT$MLQSIZ?
Read Response	$MLQSIZ: <UdpQsize>,<SmsBckupQsize>,<SmsQsize>,<TcpQsize>
Execute Syntax	N/A
Execute Response	
Unsolicited Response	N/A
Parameter Values	
<UdpQsize>	Maximum # of unsent messages stored in message log queue for UDP messages. Valid range is 0-3000. The default is 0, which is treated as a special value indicating no message size restriction. This is the equivalent of restricting the size of this queue to 3000 messages.
<SmsBckupQsize>	Maximum # of unsent messages stored in message log queue for UDP messages where SMS messaging is allowed as a backup when UDP messaging is unavailable. Valid range is 0-3000. The default is 0, which which is treated as a special value indicating no message size restriction. This is the equivalent of restricting the size of this queue to 3000 messages.
<SmsQsize>	Maximum # of unsent messages stored in message log queue for SMS messages. Valid range is 0-3000. The default is 0, which is treated as a special value indicating no message size restriction. This is the equivalent of restricting the size of this queue to 3000 messages.

AT$MLQSIZ	Message Log Queue Size
<TcpQsize>	Maximum # of unsent messages stored in message log queue for TCP messages. Valid range is 0-3000. The default is 0, which is treated as a special value indicating no message size restriction. This is the equivalent of restricting the size of this queue to 3000 messages.
Notes	All over-the-air messages sent via the event engine are first stored in flash and indexed via one of four message log queues. Due to indexing restrictions, no queue can contain more than 3000 unsent messages. For performance reasons, messages formatted via the event engine are stored in an array of 256 1000-byte flash files. Indexing for these messages is maintained in four separate flash files (one for each queue). Normally, the number of unsent messages that can be stored in flash is a function of the size of the messages. For example, when storing messages containing only NMEA data, the message log files will be completely filled after about 2500 messages. Binary messages are typically smaller and take up less space in the message log files, but are still subject to the 3000 message maximum per queue. In default operation, when the message log files are full OR when an individual queue exceeds 3000 messages, the oldest log file is deleted so that newer messages can be stored in a circular fashion using a new file in the place made available by deleting the oldest file. For some applications, the message buffers can contain far more unsent messages than is desired.

AT$MLQSIZ	Message Log Queue Size
	The purpose of this command is to mark older messages as ‘sent’ so that the number of messages in the queue never exceeds the threshold set by the user. Please note that the values specified in this command are not reflected in the value calculated for input event 61 (memory full percentage). That percentage represents the worst case (largest percentage) of either the % of the size of all unsent messages with respect to the size of all 256 log files OR the % of unsent messages with respect to the 3000 messages allowed for an individual queue. This command is used in conjunction with MSGLOGDMP and allows the user to restrict the number of unsent messages maintained in the buffers for the four message log queues: UDP, UDP with SMS backup, SMS and TCP. To Dump the message the user needs to turn off the GPS receiver with the GPSCMD.

AT$MSGLOGCL Message Log Clear

AT$MSGLOGCL	Message Log Clear
Command Function	The $MSGLOGCL command erases the log file.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$MSGLOGCL
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	None
Notes	If AT&F is performed, the message log buffer will be cleared.

AT$MSGLOGDMP Dump Unsent Messages to Serial Port

AT$MSGLOGDMP	Dump Unsent Messages to Serial Port
Command Function	This command allows the user to dump the contents of the unsent messages to the serial port. This command is non-destructive in that it does not actually remove the messages from the queue
Syntax Query	AT$MSGLOGDMP=?
Syntax	$MSGLOGDMP: (0-3),(0-1),(1-83),(0-1) OK
Write Syntax	AT$MSGLOGDMP=<queue>,<format>,<bytes_per_line>,<display_all>
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	N/A
<queue>	0 = event data that was configured to be sent to a remote server via GPRS only 1 = event data that was configured to be sent to a remote server via GPRS primarily but also use SMS as backup method if GPRS is not available 2 = event data that was configured to be sent to a remote server via SMS only 3 = event data that was configured to be sent to a remote server via TCPAPI only
<format>	0 = ASCII format (if message contains a byte that is not a printable ASCII character, it will be displayed as '?' 1 = hex format (Each byte in message is displayed as a two-digit hex character representing the value of the byte with spaces between each byte. Maximum of 16 bytes per line.)

AT$MSGLOGDMP	Dump Unsent Messages to Serial Port
<bytes_per_line>	1-83 (default = 16) number of bytes displayed per line for binary data (each byte is represented as a two-digit hex value followed by a space)
<display_all>	0 = display unsent messages only (default) 1 = display all messages (unsent and sent) from the indicated queue
Notes	This command was developed primarily as a troubleshooting utility to help debug problems related to handling unsent messages in flash. However, it has also been useful in collecting GPS data where a SIM was not available and it was not possible to connect the unit to a laptop. With this utility, you can put multiple devices in a vehicle for a drive test to collect NMEA data and dump the data for analysis easily when you return. GPS must be off to forward the data. When using larger bytes per line values (~30 and above), the user may need to issue the AT$EVTOFF=1 command to successfully dump data via this command (AT$MSGLOGDMP).

AT$MSGLOGEN Message Log Enable

AT$MSGLOGEN	Message Log Enable
Command Function	The $MSGLOGEN command has been created to enable or disable saving Event data generated via the event engine in modem's memory
Syntax Query	AT$MSGLOGEN=?
Syntax	$MSGLOGEN: (0-1) OK
Write Syntax	AT$MSLOGEN=<setting>
Write Response	OK
Read Syntax	AT$MSGLOGEN?
Read Response	$MSGLOGEN: <setting>
Execute Syntax	
Execute Response	
Unsolicited Response	N/A
Parameter Values	
<setting>	0 – 1 (possible valid values) 0 = Disable message logging (default). Event data is sent to the remote server upon occurrence. 1 = Enable message logging. Event data has to be read via AT$MSGLOGRD command or when AT$MSGLOGEN=0 is sent.
Notes	If AT$MSGLOGEN command was enabled and any unsent messages exist in memory, then the unsent data will be sent to the remote server when data logging is disabled.

AT$MSGLOGRD Message Log Read Data

AT$MSGLOGRD	Message Log Read Data
Command Function	The $MSGLOGRD command is used to query the status of the four event-engine message logs. Using the queue status this command can be used to resend messages that have already been sent and are still in the log.
Syntax Query	AT$MSGLOGRD=?
Syntax	$MSGLOGRD: (0-3),(0-x),(0-y) OK
Write Syntax	$MSGLOGRD=<queue>,<number of messages>, <starting index>
Write Response	OK
Read Syntax	AT$MSGLOGRD?

AT$MSGLOGRD	Message Log Read Data
Read Response	$MSGLOGRD: 0, 0, 0, 0, 0, 0, 0, 0,0,0 OK AT$MSGLOGRD? command returns 108 values: $MSGLOGRD: udp1,udp2,udpsms1,udpsms2,sms1,sms2,tcp1,tcp2,ussd1,ussd2 First pair of numbers: unsent UDP messages, total number of UDP messages in the UDP message log. Second pair of numbers: unsent UDP messages with SMS as backup, total number of UDP messages with SMS as backup in the UDP/SMS log. Third Pair of numbers: unsent SMS messages, total number of SMS messages in the SMS log. Fourth Pair of messages: unsent messages via TCP, total number of messages in the TCP log. Fifth pair of messages: unsent messages via USSD, total number of messages in the USSD log. Each value is comma (,) delimited. The first value of any queue represents “Total Number of Unsent Messages”. This value can be used as the <number of messages> field while resending messages. The second value of any queue represents: “Total Number of Messages Stored for that Queue”. Subtract the “Total Number of Unsent Messages” from the “Total Number of Messages Stored for that Queue” and use that as the <starting index> to resend all of the messages still in the log.
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	

AT$MSGLOGRD	Message Log Read Data
<queue>	0 –43 (possible valid values). 0 = event data that was configured to be sent to a remote server via GPRS only. 1 = event data that was configured to be sent to a remote server via GPRS primarily but also use SMS as backup method if GPRS is not available. 2 = event data that was configured to be sent to a remote server via SMS only. 3 = event data that was configured to be sent to a remote server via TCP. 4 = event data that was configured to be sent to a remote server via USSD
<number of messages>	x x = total number of messages one desires to read from the memory. A user can choose to read 1 message in which case x = 1 or read all messages in which case x = 65535.
<starting index>	y y = starting index number of messages that are stored in the memory. NOTE: y cannot be greater than total number of stored messages.
Notes	If AT&F is performed, the message log buffer will be cleared. The message logs are cleared when the modem id is changed (AT$MDMID). The message logs work on a FIFO basis. I.e. when a log is full the oldest messages are deleted to make room for new messages.

AT$MSGSND Message Send

AT$MSGSND	Message Send
Command Function	The $MSGSND command has been created to allow sending of data from one mode to another.
Syntax Query	AT$MSGSND=?
Syntax	$MSGSND: (0-5),(“ASCII DATA”) OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$MSGSND=<destination>,<”data”>
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	
<destination>	0 – 5 (possible valid values) 0 = <”data”> is sent out the serial port 1 = <”data”> is sent to all SMS addresses listed in AT$SMSDA command. 2 = <”data”> is sent via GPRS to first IP address, configured as server, in AT$FRIEND command and port number defined by AT$UDPAPI command 3 = <”data”> is sent via GPRS to IP address and Port number listed in the AT$PADDST command 4 = <”data”> is sent via GPRS to first IP address, configured as server, in AT$FRIEND command and port number for TCP API values 5 = reserved
<”data”>	a maximum of 50 bytes ASCII characters

AT$MSGSND	Message Send
Notes	AT$MSGSND command can be sent to the device via SMS, UDP-API, or serial port. AT$MSGSND=4 checks for TCPAPI usage in the first $FRIEND entry. AT$MSGSND=2 checks for UDPAPI usage in the first $FRIEND entry. ERROR will be reported if invalid usage is selected. AT$MSGSND=4 checks if TCPAPI is enabled (AT$TCPAPI=1) before attempting to send message. Will report error if not enabled. A carriage return can be added to a $MSGSND message by inserting "\0D" in to the text portion of the message For example: AT$MSGSND=0,"Hello \0D" OK Hello To add a carriage return to a $STOATEV use the following format: AT$STOATEV=1,AT$MSGSND=0,"Hello \OD"

AT$SMSDA Destination Address for SMS Messages

AT$SMSDA	Destination Address for SMS Messages
Command Function	This command allows a user to configure the phone number or email address for the sending of event data. It is also used in limiting the originating address for sending AT commands Over SMS
Syntax Query	AT$SMSDA=?
Syntax	$SMSDA: (1-5),"1234...","123.."
Write Syntax	AT$SMSDA=<index>,<dest addr>,<gateway number>
Write Response	OK
Read Syntax	AT$SMSDA?
Read Response	$SMSDA: 1,”<dest addr>”,”<gateway number>”, $SMSDA: 2,”<dest addr>”,”<gateway number>”, $SMSDA: 3,”<dest addr>”,”<gateway number>”, $SMSDA: 4,”<dest addr>”,”<gateway number>”, $SMSDA: 5,”<dest addr>”,”<gateway number>”, OK
Execute Syntax Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<index>	1 – 5 defines the index number for destination address
<dest addr>	38 characters or less phone number or email address
<gateway>	7 characters or less gateway number for email address

AT$SMSDA	Destination Address for SMS Messages
Notes	The ability to limit AT Commands over SMS is now also controlled by AT#SMSDAEN. If AT$SMSDAEN=1, (default) and if there are no addresses populated in AT$SMSDA, then all SMS with the correct AT command format addressed to the MSISDN of the device, and with the correct MDMID (if populated) will be allowed. To disable all AT Commands over SMS, set AT$SMSDAEN=0. The gateway number is provided by the Network Provider (ex: AT&T, Cingular, etc) and is only used for sending email over SMS. It is not required if you are sending SMS to a phone number. If using this command with a international number (preceded by a "+") it may be required to change the command at+csca=145. An AT$EVENT command has to be set to send a message over SMS.

.

AT$SMSDAEN Enable/Disable AT Commands over SMS

AT$SMSDAEN	Enable/Disable AT Commands over SMS
Command Function	This command allows the user to enable or disable the ability to allow AT commands over SMS
Syntax Query	AT$SMSDAEN =?
Syntax	$SMSDAEN: (0-1) OK
Write Syntax	AT$SMSDAEN =<mode>
Write Response	OK
Read Syntax	AT$SMSDAEN?
Read Response	AT$SMSDAEN: <mode> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
< mode >	0 = Disables ability to send AT commands over SMS 1 = Enables the ability to send AT commands over SMS
Notes	AT$SMSDAEN is set to 1 by default. This allows SMS over AT commands to be sent to the modem. IF there is no entries in AT$SMSDA, then all SMS with the correct AT command format, and addressed to the MSISDN of the device, and with the correct MDMID (if populated) will be allowed. AT$SMSDAEN=0 will completely disable all AT COMMAND over SMS. Any SMS received, will be treated as normal SMS. To limit AT commands Over SMS to originating from a given address, ensure AT$SMSDAEN=1 and populate the desired addresses in AT$SMSDA.
Example:	See App note GSM0308AN001- Sending AT Commands over SMS

AT$SNDMSG Send Test message

AT$SNDMSG	Send Test message
Command Function	This command allows the user to send the requested test message to the destination IP and port as defined in AT$FRIEND and AT$UDPAPI.
Syntax Query	N/A
Syntax	N/A
Write Syntax	AT$SNDMSG=<test message select >
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<test message select >	AND selected HEX options into a single 16 bit word. 01 = Send Remote Ack Test Msg 02 = Send Remote Broadcast Test Msg 04 = Send Remote Fire & Forget Test Msg 08 = Send Local PAD Test Msg 10 = Send Local UDP Test Msg
Notes	N/A

AT$UDPMSG Send and Receive UDP Messages

AT$UDPMSG	Send and Receive UDP Messages
Command Function	This command allows the user to send UDP/IP data packets while in AT command mode. The destination IP address is set by the $friend command while the port number is set by the $udpapi command. The modem must have a GPRS context activation established ($areg=2 command setting). Incoming messages addressed to the modem's IP and port specified in AT$UDPAPI will be displayed on the serial port with the unsolicited response $UDPMSG: followed by the message.
Syntax Query	AT$UDPMSG=?
Syntax	(0-1),(0-2),("data"),(0-1) OK
Write Syntax	AT$UDPMSG=<format>,<type>,<data>, <flash> <cr>
Write Response	OK
Read Syntax	AT$ UDPMSG?
Read Response	OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<format>	0 = <data> is an ASCII string (i.e.: "is this is my data") 1 = <data> is an ASCII-Hex bytes (i.e.: 050a25)

AT$UDPMSG	Send and Receive UDP Messages
<type>	0 = message will only be sent to the first IP address in the friend's list and to port number mentioned by the $UDPAPI command 1 = message will be sent via the ACK method (controlled by $ACKTM command) to the IP address listed in $FRIEND and port number listed by $UDPAPI command 2 = message will be sent to all IP address in $FRIEND command at port number listed by $UDPAPI command.
<data>	"ABCD" (Data to be transmitted in quotes) (NOTE: HEX format data shall always be entered as two ASCII characters per byte. ex: 0x5 should be entered as 05)
<flash>	This optional parameter, when enabled, allows a user to store the message in the device's FLASH memory when the device is out of coverage or not registered. Stored messages will be transmitted when the device enters GSM/GPRS coverage. 0 – Do not store messages in FLASH memory. 1 – Store messages in FLASH memory.

AT$UDPMSG	Send and Receive UDP Messages
Notes	Data received from OTA shall be sent to the modem's serial port as: $UDPMSG: <text> (ASCII or Binary data) (NOTE: Binary message will be displayed as two ASCII Hex characters <data> field from the at$udpmsg command will be sent to IP address(es) listed in the $FRIEND command and at port number defined by $UDPAPI command. <data> sent or received OTA shall be appended with a 4-byte UDP-API header as follows: Bytes 0 - 1: First 2 bytes of <data> field Byte 2: 0x06 for ASCII data type or 0x07 for Binary data type Byte 3: reserved Byte 4 - n: <data> minus the first two bytes * A minimum of 2 and maximum of 250 ASCII characters are support. For HEX, a minimum of 2 and maximum of 125 bytes are supported.

AT$WAKEUP Modem to Server Wakeup/Keep Alive

AT$WAKEUP	Modem to Server Wakeup/Keep Alive
Command Function	This command allows the user to configure the modem wakeup/keep alive parameters. These parameters control how the modem initiates contact with its server friends. Parameters can be selected so that a wakeup message sequence is executed every time the modem receives a new IP, and/or after a requested period has passed since the previous wakeup sequence has completed. A wakeup message sequence consists of sending <max retry> messages to each server friend in sequence (i.e. server 2 is contacted after all retries for server 1 is complete) and is complete when each server friend has received <max retry> messages, or upon receipt of an acknowledge message from a server.
Syntax Query	AT$WAKEUP=?
Syntax	$WAKEUP: (0-2),(0-10080) OK
Write Syntax	AT$WAKEUP=<wakeup mode>,<retry period>
Write Response	OK
Read Syntax	AT$WAKEUP?
Read Response	$WAKEUP: <wakeup mode>,<retry period>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<wakeup mode>	0 = No wakeup messages sent 1 = Send one message upon receipt of new IP and every <retry period> minutes 2 = send acknowledgement message using at$acktm parameters upon receipt of new IP and every <retry period> minutes message
<retry period >	The number of minutes for keep alive period. Zero indicates no retries.

AT$WAKEUP	Modem to Server Wakeup/Keep Alive
Notes	When this command is used, it will generate event group 0 events in the event table when the AT$EVENT? command is issued. The <retry period> parameter of this command populates the event timer value when the AT$EVTIM4? command is issued. The AT$EVTIM value will be in seconds. The parameter will also generate additional event group 0 entries. If AT$EVDEL=0 is issued or any entry for group 0 is deleted, this command MUST be re-entered for proper functionality. If a read command is issued, it will not reflect the true state of the AT$WAKEUP setting. Wakeup messages are sent to the IPs specified in AT$FRIEND and to the port specified in AT$UDPAPI command.

AT%SNCNT Query (or Reset) the Byte Counters (GPRS Only)

AT%SNCNT	Query (or Reset) the Byte Counters. (GPRS Only)
Command Function	Returns (or resets) the byte counts of every current connection.
Syntax Query	AT%SNCNT=?
Syntax	%SNCNT: (0) OK
Write Syntax	AT%%SNCNT=<rst>
Write Response	OK
Read Syntax	AT%SNCNT?
Read Response	OK %SNCNT: <nsapi1>,<upo>,<dno>,<upp>,<dnp><CR><LF> %SNCNT: <nsapi2>,<upo>,<dno>,<upp>,<dnp><CR><LF>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<rst>	resets the counters if rst = 0
<nsapi>	connection id
<upo>	uplink octets count.
<dno>	downlink octets count.
<upp>	uplink packets count.
<dnp>	downlink packets count.
Notes	If multiple contexts are activated / deactivated, set/query will result in OK followed by byte counts (set command resets the counters to 0) for each connection.

API Commands

TCP API Commands

AT$TCPAPI TCP API Control

AT$TCPAPI	TCP API Control
Command Function	This command allows the user to initiate and terminate and query the status of the TCP API connection. Please note that the TCP API can only be used over the air.
Syntax Query	AT$TCPAPI=?
Syntax	$TCPAPI: (0-1) OK
Write Syntax	AT$TCPAPI=<Status>
Write Response	OK
Read Syntax	AT$TCPAPI?
Read Response	$TCPAPI: <Status> (M-<Mgr Task>,R-<Rec Task>,T-<Trans Task>,Idx <Friend Index>)
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<Status>	TCP API connection status 0=Disabled 1=Enabled

AT$TCPAPI	TCP API Control
<Mgr Task>	TCP API Manager Task 0 = None 1 = Init 2 = Idle 3 = Connecting 4 = Connected 5 = Disconnecting
<Rec Task>	TCP API Receive Task 0 = None 1 = Init 2 = Idle 3 = Connecting 4 = Waiting for Header 5 = Waiting for Frame
<Trans Task>	TCP API Transmit Task 0 = None 1 = Init 2 = Idle 3 = Connected 4 = Sending
<Friend Index>	Friend Index (1 – 10)
Notes	N/A

AT$TCPIDLETO TCP API Idle Timeout

AT$TCPIDLETO	TCP API Idle Timeout
Command Function	Specifies the number of seconds without data traffic, in either direction, before closing the connection.
Syntax Query	AT$TCPIDLETO=?
Syntax	$TCPIDLETO: (10-86400) OK
Write Syntax	AT$TCPIDLETO=<Timeout>
Write Response	OK
Read Syntax	AT$TCPIDLETO?
Read Response	$TCPIDLETO: <Timeout>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<Timeout>	TCP API idle timeout value in seconds
Notes	After closing the connection, the device will attempt to reconnect using the FRIEND list.

AT$TCPRESTRT TCP API Restart

AT$TCPRESTRT	TCP API Restart
Command Function	If a connection exists, it is dropped and a new connection is attempted starting at the beginning of the Friend list.
Syntax Query	AT$TCPRESTRT=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$TCPRESTRT
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	N/A

AT$TCPRETRYTO TCP API Retry Timeout

AT$TCPRETRYTO	TCP API Retry Timeout
Command Function	Specifies the number of seconds without receiving a TCP level ACK that will cause the connection to be closed.
Syntax Query	AT$TCPRETRYTO=?
Syntax	$TCPRETRYTO: (120-65535) OK
Write Syntax	AT$TCPRETRYTO=<Timeout>
Write Response	OK
Read Syntax	AT$TCPRETRYTO?
Read Response	$TCPRETRYTO: <Timeout>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<Timeout>	TCP API retry timeout value in seconds
Notes	After closing the connection, the device will attempt to reconnect using the FRIEND list. The purpose of this command is to provide an abort to the TCP stack level retries. Currently, the number of retries is 10 and the amount of time varies based on calculated round trip time. The minimum time allowed is 120 seconds. Attempts to set the retry timeout to a value less than 120 or more than 65535 will result in an error.

AT$TCPSRC TCP API Source Ports

AT$TCPSRC	TCP API Source Ports
Command Function	Specifies the TCP API source port range used when making a TCPAPI connection.
Syntax Query	AT$TCPSRC=?
Syntax	$TCPSRC: (1024-65535),(1024-65535) OK
Write Syntax	AT$TCPSRC=<Start Port Number>,[<End Port Number>]
Write Response	OK
Read Syntax	AT$TCPSRC?
Read Response	$TCPSRC: <Start Port Number>,<End Port Number>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<Start Port Number>	TCP API starting port number
<End Port Number>	TCP API ending port number
Notes	Each connection attempt uses the next port number in sequence until the end port is passed. When this happens the port is set to the start port number. This current port number in use is retained over a power cycle. If only the start port number is provided, the end port number will be start port number + 49 (range of 50). The default start port number is 1024. The modem will increment the port number with each connection attempt until it reaches the default end port number of 65535. Once the modem reaches the end port number it will begin incrementing the ports again using the start port number of 1024.

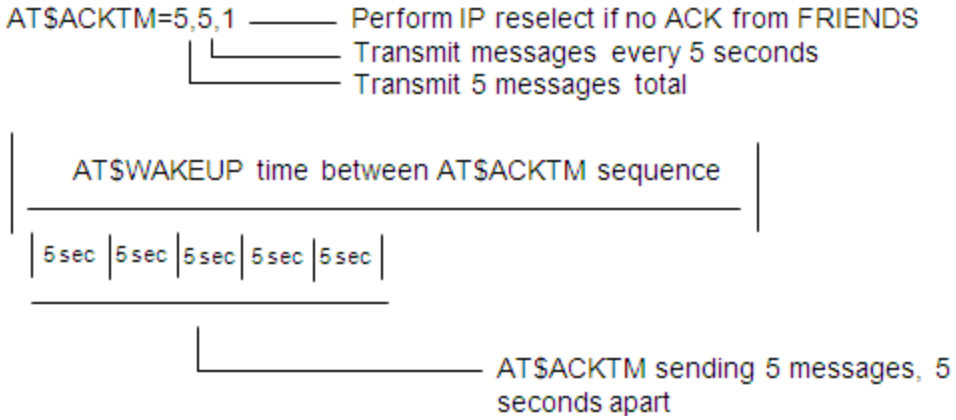
AT$TCPSTATS TCP API Statistics

AT$TCPSTATS	TCP API Statistics
Command Function	Displays bytes transmitted and received since last reset or last AT$TCPSTATS=0 command.
Syntax Query	AT$TCPSTATS=?
Syntax	$TCPSTATS: (0) OK
Write Syntax	AT$TCPSTATS=<Clear>
Write Response	OK
Read Syntax	AT$TCPSTATS?
Read Response	$TCPSTATS: Rx <Rx Bytes>,Tx <Tx Bytes>,M <Mode Change>,D <GPRS Deactivate>,R <Restarts>,C <Connection Timeout>,I <Idle Timeout>,S <Socket Errors>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<Rx Bytes>	TCP API bytes received
<Tx Bytes>	TCP API bytes transmitted
<Mode Changes>	Mode change (AT$TCPAPI=0)
<GPRS Deactivate>	GPRS deactivate
<Restarts>	TCP API restarts (AT$TCPRESTRT)
<Connection Timeout>	TCP API connection timeout
<Idle Timeout>	TCP API idle timeout
<Socket Errors>	TCP API socket errors
Notes	AT$TCPSTATS=0 will clear all TCP API statistics.

UDP API Commands

AT$ACKTM Acknowledgment Message Period & Retry Number

AT$ACKTM	Acknowledgment Message Period & Retry Number
Command Function	This command allows the user to configure the modem msg acknowledge behavior. If server acknowledgement is selected for a message, the message will be re-sent every <retry period> number of seconds until the acknowledge message sequence is complete, or until an acknowledge message is received from a server. An acknowledge message sequence consists of sending <max retry> messages to each server friend in sequence (i.e. server 2 is contacted after all retries for server 1 is complete) and is complete when each server friend has received <max retry> messages, or upon receipt of an acknowledge message from a server.
Syntax Query	AT$ACKTM=?
Syntax	$ACKTM: (0-255),(0-3600),(0,1) OK
Write Syntax	AT$ACKTM=<max retry>,<retry period>,<IP reselect>
Write Response	OK
Read Syntax	AT$ACKTM?
Read Response	$ACKTM: <max retry>,<retry period >,<IP reselect>
Execute Syntax	N/A
Execute Response	
Unsolicited Response	N/A
Parameter Values	

AT$ACKTM	Acknowledgment Message Period & Retry Number
<max retry>	The maximum number of times an acknowledge message is re-sent to a single friend server. After all retries to the friend server are exhausted, the modem will move on to the next friend server if one exists. If there are no more friend servers available, the modem will start PDP activation recovery if the recovery option is selected; otherwise, the message will be discarded. In the case of the default acknowledge wakeup message: The maximum number of wakeup messages the modem will send to each server friend upon receipt of a new IP, or upon expiration of each keep-alive period. Zero indicates no wakeup message should be sent.
<retry period >	The number of seconds between successive message retries. Zero indicates no retries.
<IP reselect >	0 = IP reselection is OFF. 1 = If an acknowledge message has not been received after all friend servers and retries for the message are exhausted, assume a problem with round-trip communication and initiate IP re-selection.
Notes	
Examples	

AT$UDPAPI Modem API Address

AT$UDPAPI	Modem API Address
Command Function	This command allows the user to query/set the API IP address and port number. Any UDP packet received from a local host and addressed to the modem API IP and port will be intercepted and processed as a modem API request. Any UDP packet received from a remote server and addressed to the modem API port will be intercepted and processed as a modem API request.
Syntax Query	AT$UDPAPI=?
Syntax	$UDPAPI: "(0-255).(0-255).(0-255).(0-255)",(0-65535) OK
Write Syntax	AT$UDPAPI=”<API IP>”,<API port>
Write Response	OK
Read Syntax	AT$UDPAPI?
Read Response	$UDPAPI: "<<API IP>>",<API port>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<API IP>	IP address for local API access
<API port >	UDP port number for local and remote API access
Notes	N/A

FOTA Commands

AT$FOTAUPG Upgrade Current Firmware

AT$FOTAUPG	Upgrade Current Firmware
Command Function	This function is used to initiate the FOTA upgrade process. It assumes that a binary delta file exists in the FFS with the name “current_delta.bin”. This file will typically be moved to the FFS via FTP functions or other FOTA specific functions.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$FOTAUPG
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Refer to Enfora Application Note ENF000AN002 for more information on FOTA operations
Example	N/A

AT$FOTACFG Configure Firmware Upgrade

AT$FOTACFG	Configure Firmware Upgrade
Command Function	This command allows the user to configure the modem Firmware Over The Air (FOTA) Server.
Syntax Query	AT$FOTACFG=?
Syntax	$FOTACFG: "host",(0-65535),"username","password",(0,1),(0-20),(0,1),(0,1),(0,1) OK
Write Syntax	AT$FOTACFG=<”ftpServerHostname”>,<ftpPort>,<”ftpUsername”>, <”ftpPassword”>,<ftpMode>,<retries>,<reportMode>,<upgradeMode>, <upgradeStatus>
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<ftpServerHostname>	Either Fully Qualified Domain Name (FQDN) or a dotted-decimal IP address of the FTP server. The FQDN is resolved by DNS immediately prior to opening IP connection to FTP server. The FQDN may be up to a maximum of 63 characters. Default value: none.
<ftpPort>	The TCP port number on the FTP server. Port range: 0 – 65535, Default value: 21
<ftpUsername>	FTP login name on the FTP server. The username may be up to a maximum of 31 characters. Default value: none.
<ftpPassword>	The FTP password for the FTP user. The password may be an empty string if the FTP user does not have a password. The password may be up to a maximum of 31 characters. Default value: none.

AT$FOTACFG	Configure Firmware Upgrade
<ftpMode>	The FTP mode to use for data connections. FTP data connections may be active or passive. Passive mode is generally better to use with FTP servers located behind firewalls. Values: 0 = active, 1 = passive. Default value: 0 (active mode).
<retries>	This is the maximum number of FTP file resumes to attempt before giving up on original AT$FOTAGET command. The range of retries is 0 – 20. Default Value: 0.
<reportMode>	The reporting mode is used to control the level of status reporting during the FOTA process. Normal reporting mode sends only critical status updates, extended reporting mode sends informative and critical status updates. The command parameters are: 0 = normal reporting, 1 = extended reporting. Default value: 0 (normal reporting mode).
<upgradeMode>	The system behavior following successful FOTA file transfer to modem. The upgrade Mode may be either Automatic or Manual. Auto mode will immediately initiate FOTA upgrade when file transfer completes. Manual mode will wait after file transfer until an explicit upgrade command is issued (AT$FOTAUPG). The command line parameters are: 0 = auto, 1 = manual. Default value: 0 (automatic mode).
<upgradeStatus>	Controls the generation of FOTA Finish status. FOTA Finish status is reported after the modem reboots from a FOTA upgrade or as a result of a fatal error. This status is only returned OTA. The command line parameters are 0 = disabled and 1 = enabled. FOTA finish status is disabled by default.
Notes	Refer to Enfora Application Note ENF000AN002 for more information on FOTA operations

AT$FOTAGET Get Firmware Upgrade OTA

AT$FOTAGET	Get Firmware Upgrade OTA
Command Function	This command will initiate a connection to the FOTA Server to download the FOTA Delta File. The FOTA Server is defined with the $FOTACFG command.
Syntax Query	AT$FOTAGET=?
Syntax	$FOTAGET: "remotefile" OK
Write Syntax	AT$FOTAGET=<"remotefile">
Write Response	OK
Read Syntax	AT$FOTAGET?
Read Response	$FOTAGET: <fotaState>,<"remotefilename">,<retriesRemaining>, <appErrorCode>,<errorCode>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<fotaState>	Current FOTA mode indicating if FOTA is in the process of transferring a FOTA file. The values are: 0 = IDLE, 1 = BUSY.
<retriesRemaining>	The number of FTP resumes remaining before terminating file transfer. This number starts with the retries specified in the AT$FOTACFG command and is decremented for each FOTA restart which occurs.
Notes	Refer to Enfora Application Note ENF000AN002 for more information on FOTA operations

AT$UPSTATUS Status of Last Upgrade

AT$UPSTATUS	Status of Last Upgrade
Command Function	This function maybe used after the FOTA upgrade process. If the upgrade process was successful it will return 00000000.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$UPSTATUS
Execute Response	00000000 OK
Unsolicited Response	N/A
Parameter Values	
<option>	N/A
Notes	

FTP Commands

AT$ATEXEC Execute AT Command File

AT$ATEXEC	Execute AT Command File
Command Function	This command is used to execute an AT command file that is stored in FFS.
Syntax Query	AT$ATEXEC =?
Response	$ATEXEC: "filename", (0-2) OK
Write Syntax	AT$ATEXEC =<“FILENAME >,<OPTION>
Response	OK
Read Syntax	AT$ATEXEC?
Response	$ATEXEC: <“FILENAME”>,<STATUS> OK
Execute Syntax	N/A
Response	N/A
Parameter Values	
<"FILENAME">	Name of file to execute
<OPTION>	AT command execution option 0 = Run To Completion 1 = Stop on Error 2 = Stop on Error and Reset Modem

AT$ATEXEC	Execute AT Command File
<STATUS>	Status from Executing File 0 = Ok 1 = Error 2 = Open File Error 3 = File Read Error 4 = AT Command Error 5 = Busy 6 = Command Length Exceeded 7 = Retries Exceeded
Notes	AT$FTPGET or AT$FFS can be used to store files into the FFS.

AT$FFS Generic Flash File System Access

AT$FFS	Generic Flash File System Access
Command Function	This command provides generic file read and write functionality to the embedded FFS in the Enfora modem.
Syntax Query	AT$FFS=?
Response	$FFS: (0-6,86),"filename",(0-524287),(0-150),("ASCII-HEX")
Write Syntax	AT$FFS=<op>,<filename>,<seek>,<length>, <data>

AT$FFS	Generic Flash File System Access
Response	Response varies by <op> Write: OK Read: $FFS: <data> OK Delete: OK Directory: $FFS [<filename>] [<filename>] […] OK Directory and size: $FFS: [<file size> <filename>] [<file size> <filename>] […] Free and used space: $FFS: <free_bytes> Free Bytes, <used_bytes> Used Bytes

AT$FFS	Generic Flash File System Access
	Delete All: OK
Read Syntax	N/A
Response	N/A
Execute Syntax	N/A
Response	N/A
Parameter Values	
<op>	FFS Operation 0 = write 1 = read 2 = delete 3 = directory 4 = directory with file size 5 = FFS free space and used space 6 = Output flash device name 86 = delete all files
<filename>	The filename in double quotes (i.e. “test.txt”) Maximum length is 16 characters. Filename parameter is required for the write (0), read (1), and delete (2) operations.
<seek>	The position in the file to read or write given as a byte offset. Seek parameter is required for the write (0) and read (1), operations. Allowed values range from 0 to 524287.
<length>	The number of bytes to read or write. Length parameter is only required for read and write operations. Allowed values range from 0 to 150.

AT$FFS	Generic Flash File System Access
<data>	Only applicable to write operations as an input parameter. Also defines the output data format for the read operation. Limited to 150 bytes (300 characters). Data is encoded in ASCII hex (i.e. 0x12, 0x34, 0xab would be entered as "1234ab"
Notes	

AT$FFS	Generic Flash File System Access
Examples	AT$FFS=0,"test",0,8,"0102030405060708" OK AT$FFS=1,"test",0,8 $FFS: 0102030405060708 OK AT$FFS=3 $FFS: test OK AT$FFS=4 $FFS: 8 test OK AT$FFS=5 $FFS: 675308 Free Bytes, 76324 Used Bytes OK AT$FFS=2,"test" OK AT$FFS=86 OK

AT$FFS	Generic Flash File System Access

AT$FTPABORT Aborts Current Data Operation with FTP Server

AT$FTPABORT	Aborts Current Data Operation with FTP Server
Command Function	Interrupts the current data operation with the FTP server.
Syntax Query	AT$FTPABORT=?
Response	OK
Write Syntax	N/A
Response	N/A
Read Syntax	AT$FTPABORT?
Response	ERROR
Execute Syntax	AT$FTPABORT
Response	OK
Parameter Values	N/A
Notes	If a data operation is not in operation, no action is performed. If a connection to an FTP server is not active, FTP INVALID CLIENT is returned.

AT$FTPBLK Change Current FTP BLock Size

AT$FTPBLK	Change Current FTP Block Size
Command Function	Changes the current FTP block size
Syntax Query	AT$FTPBLK=?
Response	$FTPBLK: (128-1500) OK
Write Syntax	AT$FTPBLK=<blocksize>
Response	OK
Read Syntax	AT$FTPBLK?
Response	1500
Execute Syntax	N/A
Response	N/A
Parameter Values	
<blocksize>	FTP block size (128-1500)
Notes	The default blocksize is 1500. AT$FTPBLK should be set prior to the AT$FTPOPEN for the blocksize to have effect.

AT$FTPCFG Configure FTP parameters

AT$FTPCFG	Configure FTP parameters
Command Function	Configures FTP parameters required to open connection to FTP Server
Syntax Query	AT$FTPCFG=?
Syntax	AT$FTPCFG: “host”, (0-65535), “username”,”password” OK
Write Syntax	AT$FTPCFG=<host>,<portnum>,<username>,<password>
Write Response	OK
Read Syntax	AT$FTPCFG?
Read Response	AT$FTPCFG: <host>, <port>,<username>,<password>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<host>	Fully qualified domain name or dotted-decimal notation IP address of FTP server, character string up to 63 characters
<portnum>	FTP server port number, 0 – 65535
<username>	FTP server user name, character string up to 31 characters
<password>	FTP server password, character string up to 31 characters
Notes	AT$FTPCFG must be entered initially before opening connection to FTP server. Configuration information will be saved to flash when flash write command is executed and will be re-used on subsequent FTP connections.

AT$FTPCHDIR Change current working directory

AT$FTPCHDIR	Change current working directory
Command Function	Changes the current working directory on the FTP server.
Syntax Query	AT$FTPCHDIR=?
Syntax	$FTPDIR: “directory” OK
Write Syntax	AT$FTPCHDIR=<directory>
Write Response	OK
Read Syntax	AT$FTPCHDIR?
Read Response	ERROR
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<directory>	Directory name on remote FTP server
Notes	AT$FTPCHDIR can only be entered after the connection to the FTP server has been established. To navigate up a directory, use the “..” directory notation.

AT$FTPCLOSE Closes the Connection to the FTP Server

AT$FTPCLOSE	Closes the Connection to FTP Server
Command Function	Closes the current connection with the remote FTP server.
Syntax Query	AT$FTPCLOSE=?
Response	OK
Write Syntax	N/A
Response	N/A
Read Syntax	AT$FTPCLOSE?
Response	ERROR
Execute Syntax	AT$FTPCLOSE
Response	OK
Parameter Values	N/A
Notes	This command is only available when a connection to the FTP server is active. If a connection to an FTP server is not active, FTP INVALID CLIENT is returned.

AT$FTPDIR Directory Listing

AT$FTPDIR	Directory Listing
Command Function	Prints a directory listing of the current working directory on the FTP server
Syntax Query	AT$FTPDIR=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	AT$FTPDIR?
Read Response	OK $FTPDIR: <current directory information>
Execute Syntax	AT$FTPDIR
Execute Response	OK <directory listing information>
Unsolicited Response	N/A
Parameter Values	N/A
Notes	AT$FTPDIR can only be entered after the connection to the FTP server has been established. Directory listing information may be delayed due to the latency of the data connection to the FTP server.

AT$FTPGET Retrieve a Remote File

AT$FTPGET	Retrieve a Remote File
Command Function	Retrieves a remote file from the FTP server for either output on the serial port or storage to the local flash file system.
Syntax Query	AT$FTPGET=?
Syntax	$FTPGET: “remotefile”,”localfile” OK
Write Syntax	AT$FTGET=<remotefile>,<localfile>
Write Response	OK
Read Syntax	AT$FTPGET?
Read Response	$FTPGET: <remotefile>,(SERIAL, <localfile>),<output>/<available>,<eof>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<remotefile>	File name on the remote FTP server.
<localfile>	File name to use for storage of the retrieved file in the local flash file system. Note: the local filename length is limited to 15 characters.
<output>	Number of bytes which have been transferred out the serial port
<available>	Number of bytes of file which are available for output to the serial port
<eof>	0 = file currently being transferred 1 = complete file has been transferred to FTP client
Notes	If local file is not specified, the remote file will be maintained in memory buffers until a subsequent FTP read request is performed to output a block of bytes onto the serial port. As the file is transmitted out the serial port, the FTP client will refill the memory buffers with additional file data from the FTP server.

AT$FTPOPEN Opens FTP Connection

AT$FTPOPEN	Opens FTP Connection
Command Function	Opens FTP control connection to FTP server
Syntax Query	AT$FTPOPEN=?
Syntax	$FTPOPEN: (“ACTIVE,”PASSIVE”) OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	AT$FTPOPEN?
Read Response	OK $FTPOPEN: <status>
Execute Syntax	AT$FTPOPEN=”<mode>”
Execute Response	OK <status>
Unsolicited Response	N/A
Parameter Values	
<mode>	ACTIVE or PASSIVE
	Mode is an optional parameter. If not specified, the FTP connection will be in the ACTIVE mode.
<status>	FTP CONNECTION OPEN FTP CONNECTION CLOSED FTP STACK ERROR Status will display the state of the control connection with the remote FTP server.
Notes	AT$FTPOPEN must be entered after FTP configuration information has been entered. If any FTP configuration information has been saved to flash, it will be used to open the connection to the FTP server. After the FTP open completes successfully, FTP CONNECTION OPEN will be printed, otherwise an error string will be printed.

AT$FTPPUT Store a Local File On the FTP Server

AT$FTPGET	Store a Local File On the FTP Server
Command Function	Stores a local file from the Flash File System onto the remote FTP server.
Syntax Query	AT$FTPPUT=?
Response	$FTPPUT:“remotefile”,“localfile” OK
Write Syntax	AT$FTPPUT=<remotefile>,<localfile>
Response	OK
Read Syntax	AT$FTPPUT?
Response	$FTPPUT:<remotefile>,<localfile>,<totalWrote>/<totalSize>,<eof>
Execute Syntax	N/A
Response	N/A
Parameter Values	
<remotefile>	File name on the remote FTP server.
<localfile>	File name on the local Flash File System.
<totalWrote>	Number of bytes of file which have been written to FTP Server
<totalSize>	Size of file to be transferred to FTP Server (in bytes).
<eof>	0 = file is currently being transferred 1 = complete file has been transferred to FTP Server
Notes	

AT$FTPR Outputs block of file data onto serial port

AT$FTPR	Outputs block of file data onto serial port
Command Function	Outputs the requested number of bytes of a retrieved file from the remote FTP server onto the serial port.
Syntax Query	AT$FTPR=?
Syntax	$FTPR: (0-1500) OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	A AT$FTPR?
Read Response	$FTPR: <available>
Execute Syntax	AT$FTPR=<blocksize>
Execute Response	OK
Unsolicited Response	NA
Parameter Values	
<blocksize>	Number of bytes to output to serial port. This number is limited to the current number of bytes of the remote file which have been transferred from the FTP server.
<available>	The number of bytes available for output to the serial port.
Notes	This command is only available after an FTP get operation has been initiated using the serial port as the file destination. It directs buffered file data from the FTP server to be transmitted in blocks out the serial port. This command is repetitively issued until the complete file is transmitted from the FTP server out the serial port.

AT$FTPRST Restart the Last FTP GET Operation

AT$FTPRST	Restart the Last FTP GET Operation
Command Function	Manual restart of an FTP GET file transfer from the last completed point in the file.
Syntax Query	N/A
Response	N/A
Write Syntax	AT$FTPRST=<offset>
Response	OK
Read Syntax	N/A
Response	N/A
Execute Syntax	N/A
Response	N/A
Parameter Values	
<offset>	Number of bytes into the file to resume file transfer. (Optional)
Notes	AT$FTPRST will attempt to resume the last incomplete file transfer from the last point in the file successfully read. Offset is not required unless the automatically computed offset is to be overridden. The AT$FTPRST command requires an RFC3659 compliant FTP server which implements the FTP REST command. AT$FTPRST is not designed to work over a system restart. The user's current path must contain the file prior to issuing AT$FTPRST or else the AT$FTPRST command will fail. The only time it is necessary to send the $FTPRST command is if a "FTP TIMEOUT" URC is received during a FTP transfer. The "FTP TIMEOUT" is typically received due loss of GPRS network connection for more than the FTP idle timeout, 60 seconds.

Real-Time Clock Commands

AT$RTCALRM Real Time Clock Alarm

AT$RTCALRM	Real Time Clock Alarm
Command Function	This command handles the setting and querying of the RTC alarm registers. When the alarm feature has been enabled the $EVENT engine will be invoked upon the going off. If the $RTCWAKE call is invoked following the alarm feature setup the modem will power back up automatically upon the alarm going off. The actions of these two features are mutually exclusive of each other, so one or the other will occur but not both.
Syntax Query	AT$RTCALRM=?
Syntax	$RTCALRM: (0-99),(1-12),(1-31),(0-23),(0-59),(0-59),(0-527040) OK
Write Syntax	AT$RTCALRM= <rtc_year>,<rtc_month>,<rtc_day>,<rtc_hour>,<rtc_min>,<rtc_sec>,<rtc_alarmTimeinMinutes>
Write Response	OK
Read Syntax	AT$RTCALRM?
Read Response	$RTCALRM: <rtc_enabled>,<rtc_year>,<rtc_month>,<rtc_day>,<rtc_hour>,<rtc_min>,<rtc_sec>,<rtc_alarmTimeinMinutes>” OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	Parameters are positional dependent, any parameter may be omitted with the use of the comma (‘,’) as a place holder on command line. If a parameter is omitted then the current value in the hardware is used.

AT$RTCALRM	Real Time Clock Alarm
< rtc_enabled >	Indicates if alarm is enabled or not. 1= Enabled 0= Disabled
< rtc_year >	The year on which the alarm is being set to trigger. The RTC supports years 2000-2099. The data is entered as a two digit value 0..99.
<rtc_month>	The month on which the alarm is being set to trigger. Values range from 1..12.
<rtc_day>	The day on which the alarm is being set to trigger. Values range from 1..31.
<rtc_hour>	The hour on which the alarm is being set to trigger. Values range from 0..24 for 24-Hour mode settings. NOTE: only 24-Hour mode currently supported.
<rtc_min>	The minute on which the alarm is being set to trigger. Values range from 0..59.
<rtc_sec>	The second on which the alarm is being set to trigger. Values range from 0..59.
<rtc_ alarmTimeinMinutes>	Periodic Alarm time in minutes. RTC Alarm will be reset at a period specified by this parameter.
Notes	This command is used to set the Alarm time for the RTC. Currently all time is based on 24-Hour time format. The alarm may be cleared using the command AT$RTCCLRA. This call in conjunction with the use of either the $EVENT engine or the $RTCWAKE command the user has a rich feature set of driving other events or waking the system up at a pre-determined time in the future. No checks are made for alarm time not being later than current time. AT$RTCALRM will not trigger if the alarm time occurs while the unit is resetting.

AT$RTCALRM	Real Time Clock Alarm
Examples	Following sets and alarm for 2003, October, 13th at 17:00 Hours at$rtcalrm=3,10,13,17,0,0 OK Following queries the alarm for current time, and shows that the alarm being; Enabled, for 2003, October 13th at 17:00 hours. at$rtcalrm? $RTCALRM: 01, 03, 10, 13, 17, 00, 00 OK Following call unsets alarm followed by displaying alarm time information. at$rtcclra OK at$rtcalrm? $RTCALRM: 00, 03, 10, 13, 17, 00, 00 OK

AT$RTCCLRA Real Time Clock Clear Alarm

AT$RTCCLRA	Real Time Clock Clear Alarm
Command Function	This command allows the modem to clear/disable the active RTC alarm. The alarm interrupt enable is cleared but alarm time not altered.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$RTCCLRA
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	None
Notes	This command will disable the RTC alarm while leaving the value of the last alarm time setting alone.

AT$RTCRSET RTC Report Reset State

AT$RTCRSET	RTC Report Reset State
Command Function	This command reports the reset state of the RTC following a power cycle. The command reports TRUE only if a reset occurred since last power up and last call to check it. So multiple calls will report the current status only.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$RTCRSET?
Execute Response	$RTCRSET : <reset state> OK
Unsolicited Response	N/A
Parameter Values	
<reset state>	1 - indicates that a RTC reset occurred, 0 - indicates that a RTC reset did NOT occur
Notes	This command returns the current reset state of the RTC since power-up. If multiple calls are made only current reset state is returned.

AT$RTCRSET	RTC Report Reset State
Examples	Following example shows the check for the RTC being reset since last check of reset and since last power up, with a response of True. AT$rtcrset? $RTCRSET: 1 OK Following example shows the check for the RTC being reset since last check of reset and since last power up, with a response of False. AT$rtcrset? $RTCRSET: 0 OK

AT$RTCTIME Real Time Clock Time

AT$RTCTIME	Real Time Clock Time
Command Function	This command handles the setting and querying of the RTC time registers.
Syntax Query	AT$RTCTIME=?
Syntax	$RTCTIME: (0-6),(0-99),(1-12),(1-31),(0-23),(0-59),(0-59) OK
Write Syntax	AT$RTCTIME= <rtc_wkday>,<rtc_year>,<rtc_month>,<rtc_day>,<rtc_hour>,<rtc_min>,<rtc_sec>
Write Response	OK
Read Syntax	AT$RTCTIME?
Read Response	$RTCTIME: <rtc_wkday>, <rtc_year>, <rtc_month>, <rtc_day>,<rtc_hour>, <rtc_min>, <rtc_sec>” OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	Parameters are positional dependent, any parameter may be omitted with the use of the comma (‘,’) as a place holder on command line. If a parameter is omitted then the current value in the hardware is used.
< rtc_wkday >	Current week day matching time day being set. The week day values range from 0-6, where; 0 = Sunday 1 = Monday 2 = Tuesday 3 = Wednesday 4 = Thursday 5 = Friday 6 = Saturday

AT$RTCTIME	Real Time Clock Time
< rtc_year >	The year to which the time is being set. The RTC supports years 2000-2099. The data is entered as a two digit value 0-99.
<rtc_month>	The month to which the time is being set. Values range from 1-12.
<rtc_day>	The day to which the time is being set. Values range from 1-31.
<rtc_hour>	The hour to which the time is being set. Values range from 0-24 for 24-Hour mode settings. NOTE: only 24-Hour mode currently supported.
<rtc_min>	The minute to which the time is being set. Values range from 0-59.
<rtc_sec>	The second to which the time is being set. Values range from 0-59.
Notes	This command is used to set the time for the RTC. Currently all time is based on 24-Hour time format.
Examples	at$rtctime? $RTCTIME: 01, 03, 10, 13, 14, 03, 2 OK at$rtctime=? $RTCTIME: (0..6), (0..99), (1..12), (1..31), (0..23), (0..59), (0..59) at$rtctime=1,3,10,13,14,37,50 OK

NITZ Commands

AT$RTCUPD Update RTC with NITZ

AT$RTCUPD	Update RTC with NITZ
Command Function	This command allows the user to enable/disable setting of the RTC time with either UTC or local time received in the NITZ.
Syntax Query	AT$RTCUPD=?
Syntax	$RTCUPD: (0-2) OK
Write Syntax	AT$RTCUPD =<mode>,<reset>
Write Response	OK
Read Syntax	AT$RTCUPD?
Read Response	$RTCUPD: <mode>,<update> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = disables updating the RTC. 1 = enables updating the RTC to UTC time 2 = Enables updating RTC to local time (based on the TZ in the NITZ)
<reset>	0 = Do not reset $RTCUPD <update> parameter 1 = Reset $RTCUPD <update> parameter to 0
<update>	0 = RTC has not been updated 1 = RTC has been updated
Notes	NITZ can be used to update RTC, however, when GPS gets a lock, UTC time will be used to update the RTC time.

AT%CNIV Generate URC with Network Name

AT%CNIV	Generate URC with Network Name
Command Function	This command allows the user to enable/disable the sending of an Unsolicited Response to the serial port with the network name, when a message has been received through NITZ.
Syntax Query	AT%CNIV=?
Syntax	%CNIV: (0,1) OK
Write Syntax	AT%CNIV =<mode>
Write Response	OK
Read Syntax	AT%CNIV?
Read Response	%CNIV: <mode> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
< mode >	0 = disables URC when NITZ is received 1 = enables URC when NITZ is received
Notes	None
Example:	%CNIV: "T-Mobile","T-Mobile","310260"

AT%CTZV Generate URC with Date and Time

AT%CTZV	Generate URC with Date and Time
Command Function	This command allows the user to enable/disable the sending of an Unsolicited Response to the serial port when the date and time have been updated from NITZ.
Syntax Query	AT%CTZV=?
Syntax	%CTZV: (0,1) OK
Write Syntax	AT%CTZV =<mode>
Write Response	OK
Read Syntax	AT%CTZV?
Read Response	%CTZV: <mode> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
< mode >	0 = disables URC when NITZ is received 1 = enables URC when NITZ is received
Notes	Time zone value in quarter hour increments, referenced to UTC time.
Example:	%CTZV: "07/03/19,19:58:36-20"

AT+CCLK Enable Setting and reading of RTC

AT+CCLK	Enable Setting and reading of RTC
Command Function	This command allows the user to set or read the Real Time Clock.
Syntax Query	AT+CCLK=?
Syntax	+CCLK: ("yy/MM/dd,hh:mm:ss+zz") OK
Write Syntax	AT+ CCLK ="yy/MM/dd,hh:mm:ss+zz"
Write Response	OK
Read Syntax	AT+ CCLK?
Read Response	+CCLK: "00/01/12,05:44:53+00" OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	Year/month/day,hour:minutes:seconds+t ime zone
Notes	+CCLK shares the same functionality as AT$RTCTIME. When one is updated, the other will also be updated. Time zone is in quarter hour increments referenced to UTC time.

AT+CSQ Signal Quality and Bit Error Rate

AT+CSQ	Signal Quality and Bit Error Rate
Command Function	Execution command returns received signal strength indication <rssi> and channel bit error rate <ber> from the ME.
Syntax Query	AT+CSQ=?
Syntax	+CSQ: 2-31,(99),(99) OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT+CSQ
Execute Response	+CSQ: <rssi>,<ber> OK
Unsolicited Response	NA
Parameter Values	
<rssi>	0 = -113 dBm or less 1 = -111 dBm 2-30 = -109... -53 dBm 31 = -51 dBm or greater 99 = not known or not detectable
<ber>	(in percent) 0-7 = as RXQUAL values in the table in GSM 05.08 [20] subclause 8.2.4 99 = not known or not detectable
Notes	N/A

RX Level	Range	
	(min) dBm	(max) dBm
0	<–110	-110
1	-110	-109
2	-109	-108
3	-108	-107
4	-107	-106
5	-106	-105
6	-105	-104
7	-104	-103
8	-103	-102
9	-102	-101
10	-101	-100
11	-100	-99
12	-99	-98
13	-98	-97
14	-97	-96
15	-96	-95
16	-95	-94
17	-94	-93
18	-93	-92
19	-92	-91
20	-91	-90
21	-90	-89
22	-89	-88
23	-88	-87
24	-87	-86
25	-86	-85
26	-85	-84
27	-84	-83
28	-83	-82
29	-82	-81
30	-81	-80
31	-80	-79
32	-79	-78

33	-78	-77
34	-77	-76
35	-76	-75
36	-75	-74
37	-74	-73
38	-73	-72
39	-72	-71
40	-71	-70
41	-70	-69
42	-69	-68
43	-68	-67
44	-67	-66
45	-66	-65
46	-65	-64
47	-64	-63
48	-63	-62
49	-62	-61
50	-61	-60
51	-60	-59
52	-59	-58
53	-58	-57
54	-57	-56
55	-56	-55
56	-55	-54
57	-54	-53
58	-53	-52
59	-52	-51
60	-51	-50
61	-50	-49
62	-49	-48
63	-48	> –48 dBm

Table: 1 - RSSI values

AT+CTZR Generate URC with Time Zone

AT+CTZR	Generate URC with Time Zone
Command Function	This command allows the user to enable/disable the sending of an Unsolicited Response to the serial port with the time zone, when a message has been received through NITZ.
Syntax Query	AT+CTZR=?
Syntax	+CTZR: (0,1) OK
Write Syntax	AT+CTZR =<mode>
Write Response	OK
Read Syntax	AT+CTZR?
Read Response	+CTZR: <mode> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
< mode >	0 = disables URC when NITZ is received 1 = enables URC when NITZ is received
Notes	Time zone value in quarter hour increments, referenced to UTC time.
Example:	None

AT+CTZU Enable Saving of Time Zone

AT+CTZU	Enable Saving of Time Zone
Command Function	This command allows the user to enable/disable the saving of the time zone information from NITZ.
Syntax Query	AT+CTZU=?
Syntax	+CTZR: (0,1) OK
Write Syntax	AT+CTZU =<mode>
Write Response	OK
Read Syntax	AT+CTZU?
Read Response	+CTZU: <mode> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
< mode >	0 = disable saving of the time zone 1 = enables saving of the time zone
Notes	None
Example:	None

Network Commands

AT$AREG Auto Registration

AT$AREG	Auto Registration
Command Function	This command sets the auto registration state of the modem
Syntax Query	AT$AREG=?
Syntax	$AREG: (0-3), (0-4) OK
Write Syntax	AT$AREG=<state>,<backoff table>
Write Response	OK
Read Syntax	AT$AREG?
Read Response	If AREG is 0-2 $AREG: <state> OK If AREG is 3 AREG: 3, 0 PDP RETRY: 15, 15, 15, 15, 60, 60, 120
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	

AT$AREG	Auto Registration
<state>	0 = Autoreg off 1 = Autoreg on 2 = Auto GPRS Activation on Power up. (for $hostif=1 and 2, MT will perform GPRS activation and go into PAD data mode. For $hostif=0 and 3, MT will perform GPRS activation, but remain in AT command mode) 3 = Auto GPRS Activation on Power up. Similar to state 2, but utilizes the $PDPBOT (PDP activation backoff time tables) to insert delay before retrying PDP activation after a PDP rejection.
<backoff table>	Selects the PDP backoff table to be used: 0 = user defined table (see $PDPBOT for display and modification of this table) 1 = Rogers Table. The delay values for this table: 240, 240, 240, 240, 1800, 3600 2 = AT&T Table. The delay values for this table: 60, 60, 60, 120, 120, 120, 180 3 = Jasper Table. The delay values for this table: 60, 60, 60, 60, 600, 900, 1800, 3600 4 = T-Mobile Table. The delay values for this table: 15, 15, 15, 15, 30, 30, 30, 30, 30, 60

AT$AREG	Auto Registration
Notes	This command sets GMS registration state. When set to 1, upon power on, the modem will automatically register on the GSM network. To set the modem to automatically attach to the GPRS network on power on, see AT%CGAATT command. AT+CGDCONT must be entered and saved before MT is placed in AREG=2 or AREG=3. If PIN is enabled, the modem will not complete the auto registration process until after the PIN has been entered (AT+CPIN). Do not use AT$AREG=2 or AT$AREG=3 with autobauding of the serial port and PAD functions. The serial port will not respond to at commands if the modem establishes a connect state before the baud rate has been determined for the serial port. The second parameter of the $AREG command is only allowed to be entered if the first parameter is 3. For the query, the second parameter is only shown if the first parameter is 3. The PDP RETRY: line shows the PDP backoff times that are currently selected.

AT$CGEER Get PDP Context Activation Reject Cause

AT$CGEER	Get PDP Context Activation Reject Cause
Command Function	This command is used to get the last GPRS PDP context activation reject cause.
Syntax Query	AT$CGEER=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$CGEER
Execute Response	$CGEER: <reject cause> OK
Unsolicited Response	NA
Parameter Values	N/A
<reject cause>	no PDP reject cause insufficient resources missing or unknown APN unknown PDP address or PDP type user authentication failed activation rejected by GGSN activation rejected, unspecified service option not supported requested service option not subscribed service option temporarily out of order NSAPI already used protocol errors

AT$CGEER	Get PDP Context Activation Reject Cause
Notes	Default reject cause is “no PDP reject cause” . <reject cause> is reset to this default reject cause by PDP context activation confirmed or PDP context deactivation confirmed.

AT$HBRST Automatic Modem Reset

AT$HBRST	Automatic Modem Reset
Command Function	This command allows the user to program the reset interval and enable/disable ignition resets on supported devices.
Syntax Query	AT$HBRST=?
Syntax	$HBRST:(0-168),(0-1*),(0-1),(0-1*) OK
Write Syntax	AT$HBRST=<hours>,<ign rst inhibit>,<motion>,<ignition>
Write Response	OK
Read Syntax	AT$HBRST?
Read Response	$HBRST:<hours>,<ign rst inhibit>,<motion>,<ignition> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<hours>	0 = Automatic reset turned off 1-168 = Number of hours until the modem resets
<ign rst inhibit>	0 = inhibit off (default). Modem will be reset when ignition on event is detected. 1 = Inhibit reset upon ignition on Note: If the device is being reset by ignition input (switched power), the frequency of the heartbeat reset can be set as high as 168 hours. If ignition resets are inhibited, the maximum heartbeat reset allowed is 24 hours. Therefore, if <ign rst inhibit> = 1, <hours> must = (1-24) * Note: On devices that do not support ignition detection, this parameter value can only be set to 1.

AT$HBRST	Automatic Modem Reset
<motion>	0 = Postpone automatic reset when <hours> threshold is reached if device is in ‘driving’ motion state until mode transitions to ‘parked’ motion state. 1= Ignore motion state when implementing automatic reset logic.
<ignition>	0 = Postpone automatic reset when <hours> threshold is reached if device is in 'ignition on' state until mode transitions to 'ignition off' state. 1 = Ignore ignition state when implementing automatic reset logic. * Note: On devices that do not support ignition detection, this parameter value can only be set to 1.
Notes	* On devices that do not support ignition detection (such as the MT 1000), the only acceptable value for the <ign rst inhibit> and <ignition> parameters is 1. The time until the modem resets is an approximate value. If the device is being reset by ignition input (switched power), the frequency of the heartbeat reset can be set as high as 168 hours. If ignition resets are inhibited, the maximum heartbeat reset allowed is 24 hours.

AT$LOCI Location Information

AT$LOCI	Location Information Configuration
Command Function	This command allows the user to enable storage of the GSM LOCI info in the modem NVMEM
Syntax Query	AT$LOCI=?
Syntax	(0-2) OK
Write Syntax	AT$LOCI=<mode> <cr>
Write Response	OK
Read Syntax	AT$ LOCI?
Read Response	$LOCI: <mode>,<IMSI>,<TMSI>,<LAI>,<TMSI Time>,<LOC UPDATE STATUS> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<fmode>	0 = GSM LOCI information is stored in the SIM 1 = GSM LOCI information is stored in the Modem 2 = GSM LOCI information is stored in the SIM if SIM data is valid. Otherwise GSM LOCI information is stored in the modem.
Notes:	The GSM LOCI is saved in non-volatile memory every time the SIM's GSM LOCI is updated. AT&W is not needed to save the settings.

AT$LUPREJ Get LUP Reject Cause

AT$LUPREJ	Get LUP Reject Cause
Command Function	This command is used to get the last Location Area Update cause.
Syntax Query	AT$LUPREJ=?
Syntax	$LUPREJ: (0,1)
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$LUPREJ?
Execute Response	$LUPREJ: <output>,<cause>,<MCC/MNC> OK
Unsolicited Response	N/A
Parameter Values	
<cause>	Location Area Update reject cause. See notes section for reject codes.
<MCC/MNC>	Mobile network that issued the Reject
Notes	

AT$LUPREJ	Get LUP Reject Cause
LUP Reject codes:	02 = RC_IMSI_IN_HLR 03 = RC_ILLEGAL_MS 04 = RC_IMSI_IN_VLR 05 = RC_IMEI_NOT_ACCEPTED 06 = RC_ILLEGAL_ME 11 = RC_PLMN_NOT_ALLOWED 12 = RC_LA_NOT_ALLOWED 13 = RC_ROAMING_NOT_ALLOWED 17 = RC_NETWORK_FAILURE 22 = RC_CONGETION 32 = RC_SERVICE_NOT_SUPPORTED 33 = RC_SERVICE_NOT_SUBSCRIBED 34 = RC_SERVICE_ORDER 38 = RC_IDENTIFIY 95 = RC_INCORRECT_MESSAGE 96 = RC_INVALID_MAND_MESSAGE 97 = RC_MESSAGE_TYPE_NOT_IMPLEM 98 = RC_MESSAGE_TYPE_INCOMPAT 99 = RC_IE_NOT_IMPLEM 100 = RC_CONDITIONAL_IE 101 = RC_MESSAGE_INCOMPAT 111 = RC_UNSPECIFIED
Examples	AT$LUPREJ $LUPREJ: 0,13,310260 Network 310260 (TMO) reject the Location Area Update for roaming not allowed

AT$MSCLS Set GPRS Multislot Class

AT$MSCLS	Set GPRS Multislot Class
Command Function	This command is used to set the GPRS multislot class.
Syntax Query	AT$MSCLS=?
Syntax	$MSCLS: (1-6, 8-10) OK
Write Syntax	AT$MSCLS=<msclass>
Write Response	OK
Read Syntax	AT$MSCLS?
Read Response	$MSCLS: <msclass> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	NA
Parameter Values	N/A
<msclass>	1-6, 8-10
Notes	The value is saved when using AT&W command. To return to default MS class, use AT&F command.

AT$NETLST Network List

AT$NETLST	Network List
Command Function	This command will display the top three channels of each network found.
Syntax Query	AT$NETLST=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$NETLST
Execute Response	$NETLST: MCC: 310 MNC: 260 LAC CellId ARFCN RSSI 27b0 3414 747 24 27b0 3413 740 18 MCC: 310 MNC: 410 LAC CellId ARFCN RSSI Cb23 7f9 232 19 Cb23 41c 160 18 Cb23 160 248 18 OK
Unsolicited Response	N/A
Parameter Values	N/A

AT$NETLST	Network List
Notes	This command may take up to 5 minutes to complete. If the modem is registered, it will deregister and remain deregistered until the AT$NETLST is complete. Also if modem has an active PDP context, the PDP context will be deactivated and remain deactivated until the AT$NETLST is complete. After the AT$NETLST is complete, the modem will re-register and re-activate the PDP context depending on user settings.

AT$NETMON Monitor Network Availability

AT$NETMON	Monitor Network Availability
Command Function	This command allows the modem to take aggressive network recovery action based upon the results of continuous network monitoring.
Syntax Query	AT$NETMON=?
Syntax	$NETMON: (0,5-1440),(0-10),(0-255),(0-3) OK
Write Syntax	AT$NETMON= <net_unavail_min>,<reset_cnt>,<ping check>,<rst timers>
Write Response	OK
Read Syntax	AT$NETMON?
Read Response	AT$NETMON: "<net_unavail_min >,<reset_cnt>,<ping check>,<rst timers>"
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<net_unavail_min>	Number of minutes the network must remain unavailable before current Network Activation is released, and a new Network Activation is attempted. Network availability is valid Network IP (AT$NETIP). A value of zero means the Network Activation will never be released via AT$NETMON.
<reset_cnt>	TBD
<ping check>	Number of minutes between modem-initiated ping checks. If no network data has been received within <ping check> minutes, the modem will initiate pings (up to 4 ICMP messages are generated) to the 1st server on the $FRIEND list. If no ping response is received to any of the 4 ICMP messages, the modem will initiate pings to the next server in the list. If no ping response is returned from any of the $FRIEND servers, a new IP is obtained via a modem-initiated GPRS de-activation / activation sequence. A value of zero indicates that the modem will never initiate a ping check.

AT$NETMON	Monitor Network Availability
<rst timers>	0 - Reset network monitoring timers upon any activity on the serial port AND use legacy IP reselect for <ping check> recovery 1 - Do not reset the network monitoring timers if there is activity on the serial ports 2 - Use modem reset for <ping check> recovery 3 - Do not reset the network monitoring timers if there is activity on the serial ports AND use modem reset for <ping check> recovery
Notes	

AT$RESET Reset Modem

AT$RESET	Reset Modem
Command Function	This command is used to perform a modem reset.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$RESET
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	N/A
Notes	Execution of this command will perform a full reset of the software stack. If the modem is currently registered onto the network, the modem will perform a detach before performing the stack reset.

AT$TCPERR TCP Error Codes

AT$TCPERR	TCP Error Codes
Command Function	This command outputs the TCP error codes.
Syntax Query	AT$TCPERR=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	AT$TCPERR?
Read Response	$TCPERR: <TcpLastError>,<TcpPadDiscReason>,<TcpApiDiscReason> OK

AT$TCPERR	TCP Error Codes
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<TcpLastError>	TCP ERROR values 0 = no error stored 1 = normal closure 2 = unexpected RST 3 = unexpected SYN 4 = received FIN 5 = timeout establishing connection 6 = failure establishing connection 7 = timeout in close 8 = timeout on established connection 9 = RST on established connection 10 = ICMP rec'd: no route to host 11 = ICMP rec'd: connection refused 12 = ICMP rec'd: frag req, DF set 13 = no route found for dest 14 = interface down or other failure 15 = internal stack failure

AT$TCPERR	TCP Error Codes
<TcpPadDiscReason>	PAD DISC values 0 = no reason stored 1 = server timeout 2 = PPP caused 3 = host caused 4 = connection timeout 5 = caused by AT cmd
<TcpApiDiscReason>	API DISC values 0 = no reason stored 4 = connection timeout 5 = caused by AT cmd 6 = bad msg format 7 = connect fail
Notes	None yet.

AT%BAND Frequency Band Information

AT%BAND	Frequency Band Information
Command Function	This command sets the Frequency bands the modem will scan for available network service.
Syntax Query	AT%BAND=?
Syntax	%BAND: (0-1),(<band>)* OK
Write Syntax	AT%BAND= <mode>,<band>
Write Response	N/A
Read Syntax	AT%BAND?
Read Response	%BAND: 0,<band>
Execute Syntax	AT%BAND
Execute Response	%BAND: <band> OK
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = automatic 1 = manual
<band>	1 = GSM 900 MHz 2 = DCS 1800 MHz 4 = PCS 1900 MHz 8 = EGSM channels (in 900 band but not all the GSM channels) 16 = 850 Note: *applicable to only the bands supported by the device.

AT%BAND	Frequency Band Information
Examples of combining Primary bands	11 = GSM/EGSM/DCS 15 = GSM/EGSM/DCS/PCS 20 = 850/PCS 31 = GSM/EGSM/DCS/PCS/850
Notes	Usable frequency bands dependent on product type. Do not enter <band> in Write command if <mode> is automatic. After entering and saving this command, the modem must be power cycled before it will take effect.
Examples	The parameter values for <band> can be added together to support multiple frequency bands. 1 + 8 = 9 – The value of 9 is a combination of adding the bands 1 and 8 together, which would include the complete 900 MHz band, supported by the Enfora radio. 1 + 2 + 4+ 8 + 16 = 31 – The combination of all values supports the quad-band radio.

AT%CGAATT Automatic Attach and Detach Mode

AT%CGAATT	Automatic Attach and Detach Mode
Command Function	This command is used to chose the behavior of the attach procedure.
Syntax Query	AT%CGAATT=?
Syntax	%CGAATT: (0,1),(0,1) OK
Write Syntax	AT%CGAATT=<att_m>,<det_m>
Write Response	OK
Read Syntax	AT%CGAATT?
Read Response	%CGAATT: 1,1 OK
Execute Syntax	
Execute Response	
Unsolicited Response	N/A
Parameter Values	
<att_m>	automatic attach mode 0 =automatic attach 1 =manual attach
<det_m>	automatic detach mode 0 =automatic detach after last context deactivation 1 =manual detach
Notes	When automatic attach/detach is enabled and at$areg is equal to 1, 2 or 3, the modem will automatically attach onto and detach from the GPRS network upon power on or power down.

AT%CGREG GPRS Extended Registration State

AT%CGREG	GPRS Extended Registration State
Command Function	This command reports extended information about GPRS registration state. %CGREG behaves exactly as +CGREG does. In addition %CGREG supports three states +CGREG does not support.
Syntax Query	AT%CGREG=?
Syntax	%CGREG: (0,3) OK
Write Syntax	AT%CGREG=<mode>
Write Response	OK
Read Syntax	AT%CGREG?
Read Response	%CGREG: <n>,<stat>,[,<lac>,<ci>,<act>] OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	enable or disable extended GPRS registration state reporting 0 = do not report registration state 1 = do report registration state 2 = enable network registration and location information unsolicited result code +CGREG: <stat>[,<lac>,<ci>] 3 = enable network registration, location information, and activated/deactivated PDP context unsolicited result code +CGREG: <stat>[,<lac>,<ci>,<act>].

AT%CGREG	GPRS Extended Registration State
<state>	0 = not registered 1 = registered to home network 2 = not yet registered, but searching for network to register to 3 = registration denied 4 = unknown state 5 = registered to foreign network (roaming) 6 = limited service (cell might be overloaded) 7 = GSM call active 8 = no cell available 9 = next attempt to update MS
<lac>	string type; two-byte location area code in hexadecimal format (e.g. "00C3" equals 195 in decimal)
<ci>	string type; two-byte cell ID in hexadecimal format
<act>	0 = deactivated 1 = activated
Notes	N/A

AT%NRG Network Registration and Service

AT%NRG	Network Registration and Service Selection
Command Function	Set command forces an attempt to select and register the GSM network operator. <regMode> is used to select whether the selection is done automatically by the ME or is forced by this command to operator <opr> (it shall be given in format <oprFrmt>).
Syntax Query	AT%NRG=?
Syntax	%NRG: (0,1,4),(0-3),(0-2) OK
Write Syntax	AT%NRG=<regMode>,<srvMode>,<oprFrmt>,<opr>
Write Response	OK
Read Syntax	AT%NRG?
Read Response	%NRG=<regMode>,<srvMode>,<oprFrmt>,<srvStat>,<opr> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<regMode>	0 = automatic registration (<opr> field is ignored) 1 = manual registration (<opr> field shall be present on registration attempt) 4 = both
<srvMode>	0 =full service 1 = limited service 2 = no service 3 = set registration mode only

AT%NRG	Network Registration and Service Selection
<oprFrmt>	0 = long format alphanumeric <opr> 1 = short format alphanumeric <opr> 2 = numeric <opr>
<srvStat>	0 = full service 1 = limited service 2 = no service
<opr>	string type
<oprFrmt>	indicates if the format is alphanumeric or numeric; long alphanumeric format can be up to 16 characters long and short format up to 8 characters; numeric format is the GSM Location Area Identification number (refer GSM 04.08 subclause 10.5.1.3) which consists of a three BCD digit country code coded as in ITU-T E.212 Annex A, plus a two BCD digit network code, which is administration specific; returned <opr> shall not be in BCD format, but in IRA characters converted from BCD; hence the number has structure: (country code digit 3)(country code digit 2)(country code digit 1)(network code digit 2)(network code digit 1)
Notes	The command %NRG is an expansion of the +COPS command. The new command allows specifying the service state of the registration. For a list of current available network operators please use the test command of AT+COPS=?

AT+CREG Network Registration Info

AT+CREG	Network Registration Info
Command Function	Write command controls the presentation of an unsolicited result code +CREG: <stat> . Read command returns the status of result code, which shows whether the network has currently indicated the registration of the ME.
Syntax Query	AT+CREG=?
Syntax	+CREG: (0,2) OK
Write Syntax	AT+CREG=[<n>]
Write Response	OK
Read Syntax	AT+CREG?
Read Response	+CREG: <n>,<stat>[,<lac>,<ci>] OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<n>	0 = disable network registration unsolicited result code 1 = enable network registration unsolicited result code +CREG: <stat> 2 = enable network registration and location information unsolicited result code +CREG: <stat>[,<lac>,<ci>]

AT+CREG	Network Registration Info
<stat>	0 = not registered, ME is not currently searching a new operator to register to 1 = registered, home network 2 = not registered, but ME is currently searching a new operator to register to 3 = registration denied 4 = unknown 5 = registered, roaming
<lac>	string type; two-byte location area code in hexadecimal format (e.g. "00C3" equals 195 in decimal)
<ci>	string type; two-byte cell ID in hexadecimal format
Notes	N/A

AT$PDPBOT PDP Activation Backoff Time

AT$PDPBOT	PDP Activation Backoff Time
Command Function	This command allows the display and editing of the user defined table for the PDP Activation Backoff Time. See $AREG=3 for a description of the use of the PDP Activation Backoff Time.
Syntax Query	AT$PDPBOT=?
Syntax	$PDPBOT: (5-7200),(5-7200),(5-7200),(5-7200), (5-7200),(5-7200),(5-7200),(5-7200),(5-7200),(30-7200) OK
Write Syntax	AT$PDPBOT=<time1>,<time2>,...<time10>
Write Response	OK
Read Syntax	AT$PDPBOT?
Read Response	$PDPBOT:<time1>,<time2>,...<time10> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<time1> thru <time10>	Delay time for PDP Activation retry after a PDP activation rejection. There can be between 1 and 10 time values. Each value must be the same or larger than the preceding value. The values are in seconds. The valid range for the values is 5 to 7200 for all but the last value; the range for the last value is 30 to 7200.
Notes	

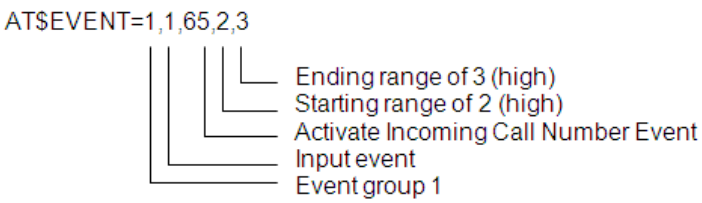
Event Commands

AT$ETSAV# Event Timer Save Configuration

AT$ETSAV#	Event Timer Save Configuration
Command Function	These commands allow the user to set/query a flag that is used to determine whether the event timer value will be persistent through a reset. If the flag is set for the timer, the timer count will be saved roughly once a minute and the saved value will be used as the starting value for the timer following a reset. This is intended to support long-range timers (for example, 3 hours) where resolution of +/-1 minute would be acceptable. If the flag is clear, the timer always starts at zero following an reset. The # sign represents timers 1 through 10 (i.e. ETSAV1).
Syntax Query	AT$ETSAV#=?
Syntax	$ETSAV#:(0-1) OK
Write Syntax	AT$ETSAV#=<flag>
Write Response	OK
Read Syntax	AT$ETSAV#?
Read Response	$ETSAV#: <flag>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<flag>	1 =save this event timer value roughly once a minute to flash 0 = restart this event timer at zero following reset (default)
Notes	AT&W must be issued after flag is set to ensure flag retains its status through a reset.
Example	N/A

AT$EVCID User Defined Incoming Call Number Event

AT$EVCID	User defined incoming call number event
Command Function	This command allows the user to define up to 5 separate incoming call number user input events
Command Functional	Enfora Specific
Group	
Syntax Query	AT$EVCID=?
Syntax	$EVCID: (0-5),(0-2),"<number" OK
Write Syntax	AT$EVCID=<entry>,<mode>[,<“number”>]
Write Response	OK
Read Syntax	AT$EVCID?
Read Response	$EVCID: <entry>,<mode>,<“number”>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<entry>	1-5 = Selects which CID entry to modify
<mode>	0 = Disable event generation for incoming call number 1 = Enable event generation for incoming call number and suppress ring indication and respond to network with busy signal. 2 = Enable event generation for incoming call number and do not suppress ring indication.
<number>	string type;Character string [~]<0..9,+,?>. Where <?> is a single character wildcard. If number starts with ‘~’ it will match to any incoming call number with 0 or more digits preceding the remaining digits in the string. This is useful for matching to local, national and international ISDN telephony numbering plans.

AT$EVCID	User defined incoming call number event
Notes	In the event the incoming call number matches more than one incoming call number selection, the mode selection will be based on priority order. The priority order will be for entries 1 through 5 with entry 1 having the highest priority.
Example:	These commands will cause the example in AT$EVENT to trigger for incoming call numbers matching event call id 2 or event call id 3. AT$EVCID=2,1,"123456789?" // Define incoming call number with the last digit a wildcard AT$EVCID=3,1,"~123456789" // Define incoming call number to allow For local and international prefixes 

AT$EVDEL Delete Event

AT$EVDEL	Delete Event
Command Function	This command allows the user to delete items from the user generated event table. Entering only the group number will delete the whole group.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$EVDEL=<group><letter ID>
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	
<group>	event list group number
<letter ID>	letter indicating which element of the group (optional)
Notes	
Example:	AT$EVDEL=1 Will delete all entries event group 1 AT$EVDEL=1b Will delete only the second entry in event group 1

AT$EVDELA Delete Event (All)

AT$EVDELA	Delete Event (All)
Command Function	This command allows the user to delete all events from the event table.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$EVDELA
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	N/A

This command deletes ALL events from the event table including the default events that control the flashing of the status LEDs.
If the default events are deleted, they must be manually recreated or use AT&F to restore the factory configuration.

AT$EVDELR Delete a Range of Event Groups

AT$EVDELR	Delete a Range of Event Groups
Command Function	This command deletes a range of event groups.
Syntax Query	AT$EVDELR=?
Syntax	$EVDELR: (0-255),(0-255) OK
Write Syntax	AT$EVDELR=<start>,<stop>
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<start>	First group index in range to be deleted
<stop>	Last group index in range to be deleted.
Notes	N/A

AT$EVENT User Defined Input/Output

<table>
<tr><th>AT$EVENT</th><th>User Defined Input/Output</th></tr>
<tr><td>Command Function</td><td>This command allows the user to customize the modem’s input and output capabilities. Any combination of input events can be monitored to trigger any combination of output events.</td></tr>
<tr><td>Syntax Query</td><td>AT$EVENT=?</td></tr>
<tr><td>Syntax</td><td>$EVENT: (0-255),(0-3),(0-255),(-2147483647 - 2147483647),(-2147483647 - 2147483647)</td></tr>
<tr><td>Write Syntax</td><td>AT$EVENT=<event group>,<event type>,<event category>,<parm1>,<parm2></td></tr>
<tr><td>Write Response</td><td>OK</td></tr>
<tr><td>Read Syntax</td><td>AT$EVENT?</td></tr>
<tr><td>Read Response</td><td>$EVENT:
<table>
<tr><th>evgp</th><th>evtyp</th><th>evcat</th><th>p1</th><th>p2</th></tr>
<tr><td>1A</td><td>0</td><td>27</td><td>1</td><td>1</td></tr>
<tr><td>1B</td><td>3</td><td>22</td><td>0</td><td>0</td></tr>
<tr><td>2A</td><td>0</td><td>27</td><td>0</td><td>0</td></tr>
<tr><td>2B</td><td>3</td><td>14</td><td>0</td><td>0</td></tr>
<tr><td>3A</td><td>0</td><td>9</td><td>2</td><td>4</td></tr>
<tr><td>3B</td><td>3</td><td>37</td><td>1</td><td>0</td></tr>
<tr><td>4A</td><td>0</td><td>9</td><td>5</td><td>5</td></tr>
<tr><td>4B</td><td>3</td><td>21</td><td>0</td><td>0</td></tr>
<tr><td>5A</td><td>0</td><td>9</td><td>0</td><td>0</td></tr>
<tr><td>5B</td><td>3</td><td>13</td><td>0</td><td>0</td></tr>
<tr><td>6A</td><td>0</td><td>9</td><td>1</td><td>1</td></tr>
<tr><td>6B</td><td>3</td><td>21</td><td>0</td><td>0</td></tr>
<tr><td>8A</td><td>1</td><td>200</td><td>1</td><td>1</td></tr>
<tr><td>8B</td><td>3</td><td>152</td><td>1</td><td>70</td></tr>
</table></td></tr>
<tr><td>Execute Syntax</td><td>N/A</td></tr>
</table>

AT$EVENT	User Defined Input/Output
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<event group>	This parameter defines the group number of a group of events and the order they are executed. Events are grouped together to control execution sequence. A group number has to have at least one input event and one output event. Multiple input events within a group number would be treated as a logical AND condition. Multiple output events within a group number would be executed individually in a sequential manner. Valid values for group number are: 0 thru 255. For additional details see the Event Tables section of this manual.
<event type>	This parameter defines the type of event: Input or Output. An Input event can be defined as: Transition, Occurrence, or Input. The output event is executed when input event conditions are met. For additional details see the Event Tables section of this manual.
<event category> , <parm1>, <parm2>	These parameters define the actual Input or Output Event number and their valid range for <parm1> and <parm2>. For additional details see the Event Tables section of this manual.

AT$EVENT	User Defined Input/Output
Notes	A maximum of 600 events (input and output) are supported. User variable values are automatically saved to flash and explicit user intervention to save the values is not required. AT&W does not save user variables to flash. At the end of each event engine cycle, if any user variable has been changed the user variable values are automatically saved to flash at that time (so you can make multiple changes to user variables inside event groups without worrying about writing to flash too often). Changing a user variable via $EVTEST also causes the user variable values to be saved to flash. If you need to downgrade your device to a package previous to package 52 and you have event groups greater than 99 and/or events totalling more than 400, then those events must be removed since firmware packages before 52 only support 0-99 event groups and only 400 events total. The non-supported events can be removed using AT$EVDEL command or by completly deleting all events using AT$EVDELA or AT&F. Contact firmware support for further details.

AT$EVGQRY Event Group Query

AT$EVGQRY	Event Group Query
Command Function	Allows user to query the contents of a single event group (same output format as $EVENT?, but limited to a single event group).
Syntax Query	AT$EVGQRY=?
Syntax	AT$EVGQRY: (0-255) OK
Write Syntax	AT$EVGQRP=<event group>
Write Response	$EVGQRY: evgp evtyp evcat p1 p2 1A 0 27 1 1 1B 3 22 0 0 OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<event group>	This parameter defines the group number of a group of events and the order they are executed. Events are grouped together to control execution sequence. A group number has to have at least one input event and one output event. Multiple input events within a group number would be treated as a logical AND condition. Multiple output events within a group number would be executed individually in a sequential manner. Valid values for group number are: 0 thru 255.
Notes	

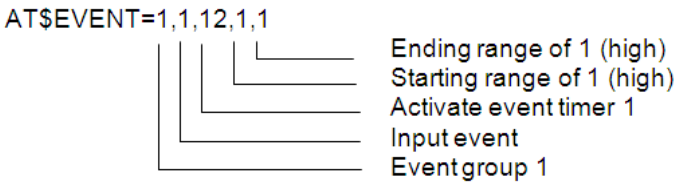
AT$EVNTRY Event Query

AT$EVNTRY	Event Query
Command Function	This command queries how many events have been used and how many are left.
Syntax Query	AT$EVNTRY=?
Syntax	ERROR
Write Syntax	N/A
Write Response	N/A
Read Syntax	AT$EVNTRY?
Read Response	$EVNTRY: <used>,<left>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<used>	Number of events that have been used
<left>	Number of events available for new entries
Notes	

AT$EVTEST Generate Test Input Event

AT$EVTEST	Generate Test Input Event
Command Function	This command allows the user to generate any input event. This is useful for testing the user event table.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$EVTEST=<event>,<state>
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	
<event>	input event number
<state>	input event test state
Notes	User variable values are automatically saved to flash and explicit user intervention to save the values is not required. AT&W does not save user variables to flash. At the end of each event engine cycle, if any user variable has been changed the user variable values are automatically saved to flash at that time (so you can make multiple changes to user variables inside event groups without worrying about writing to flash too often). Changing a user variable via $EVTEST also causes the user variable values to be saved to flash.

AT$EVTIM# User Defined Input Event Timers

AT$EVTIM#	User Defined Input Event Timers
Command Function	This command allows the user to define up to 10 separate periodic input events in 1 second increments
Syntax Query	AT$EVTIM#=?
Syntax	$EVTIM#: (0,6-604800) OK
Write Syntax	AT$EVTIM#=<rate>
Write Response	OK
Read Syntax	AT$EVTIM#?
Read Response	$EVTIM#: <rate>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<rate>	number of seconds between each generated input event.
Notes	AT$WAKEUP <retry_period> uses Event Timer 4. AT$EVTIM4=x will affect the value set by AT$WAKEUP. Do Not use AT$EVTIM4=x if using AT$WAKEUP.
Example	These commands will cause the example in AT$EVENT to trigger every 60 seconds. AT$EVTIM1=60 

AT$EVTIMQRY Event Counter

AT$EVTIMQRY	Event Counter
Command Function	This command shows the current count for the event counter of the timer specified indicated by the argument.
Syntax Query	AT$EVTIMQRY=?
Syntax	$EVTIMQRY: (1-10) OK
Write Syntax	AT$EVTIMQRY=<timer_index>
Write Response	$EVTIMQRY:<timer_index>=<count>
	OK
Read Syntax	AT$EVTIMQRY?
Read Response	ERROR
Execute Syntax	AT$EVTIMQRY=10
Execute Response	$EVTIMQRY: 10=0.000 OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	AT$EVTIMQRY=x will respond with ERROR if X timer has not been set (0).

AT$EVTOFF Event Engine Disable

AT$EVTOFF	Event Engine Disable
Command Function	The $EVTOFF command gives the user the ability to disable the event engine. For example, this would be a good command to run prior to a software upgrade.
Syntax Query	AT$EVTOFF=?
Syntax	$EVTOFF:(0-1) OK
Write Syntax	AT$EVTOFF=<state>
Write Response	OK
Read Syntax	AT$EVTOFF?
Read Response	$EVTOFF: <state>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<state>	0 = event engine enabled (default) 1 = event engine disabled
Notes	N/A
Example	N/A

AT$EVTQRY Query the State or Value of the Specified Input Event

AT$EVTQRY	Query the State or Value of the Specified Input Event
Command Function	This command allows the user to query the state or value of the input event number
Syntax Query	AT$EVTQRY=?
Syntax	$EVTQRY: (0 - <max input event>) OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$EVTQRY=<input event>
Execute Response	$EVTQRY: <input event> = state
Unsolicited Response	N/A
Parameter Values	
<max input event>	Maximum input event # supported by the current firmware build(see $EVENT for the current maximum input event value).
<input event>	Range: 0-<max input event> Selects which input event to query
Notes	
Example:	AT$EVTQRY=29 /* query input event 29 (Invalid GPS) $EVTQRY: 29=1436 /* 1436 seconds of consecutive invalid GPS data OK

AT$STOATEV Store AT Command Events

AT$STOATEV	Store AT Command Events
Command Function	This commands allows the user to store AT command output events. The AT command is executed upon the triggering of the associated input event.
Syntax Query	AT$STOATEV=?
Syntax	$$STOATEV: (1-35),<AT commands> OK
Write Syntax	AT$STOATEV = <1-35>, < AT command >
Write Response	OK
Read Syntax	AT$ STOATEV?
Read Response	$STOATEV: AT Event# AT Cmds 1 2 … … 35 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<1-35 >	AT event index.
<AT command>	AT command associated with the AT event index. The AT command is not checked for validity.

AT$STOATEV	Store AT Command Events
Notes	This command is used in conjunction with the Dynamic Input Output event (AT$EVENT). The output event associated with this command is event 44. When output event 44 is defined in the event table, Parm1 defines which index to refer to. The AT command associated with the index is executed. The use of Dynamic Event Scripting using AT$EVENT or AT$EVDEL as a stored AT Command Event can lead to unpredictable operation and is not recommended. When storing command to dial a voice call, a “v” replaces the “;” at the end of the dial string..(i.e., atd17195551212v)

AT$USRVAL User Value

AT$USRVAL	User Value
Command Function	Allows the user to store a value in flash memory which can later be retrieved.
Syntax Query	AT$USRVAL=?
Syntax	$USRVAL: (0-FFFFFFFF) OK
Write Syntax	AT$USRVAL=<hex value>
Write Response	OK
Read Syntax	AT$USRVAL?
Read Response	$USRVAL:(hex value) OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<hexval>	(0-FFFFFFFF)
Notes:	N/A

SIM Toolkit Commands

AT%SATC SET SIM Application Toolkit Configuration

AT%SATC	SET SIM Application Toolkit Configuration
Command Function	This command sets the configuration for SIM application toolkit download mechanism.
Syntax Query	AT%SATC=?
Syntax	%SATC: (0,1),(40) OK
Write Syntax	AT%SATC=<n>,<satPrfl>
Write Response	OK
Read Syntax	AT%SATC?
Read Response	SATC: =<n>,<satPrfl > OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<n>	0 = disable presentation of unsolicited notifications result codes from the TA to the TE 1 = enable presentation of unsolicited notifications result codes from the TA to the TE
<prflLen>	Length in Bytes of the current <satPrfl>
<satPrfl>	String type: SIM application toolkit profile, starting with the first byte of the profile.

AT%SATC	SET SIM Application Toolkit Configuration
Notes	Associated commands: AT%SATT,AT%SATE, AT%SATR. Associated results codes %SATE, %SATA,%SATN and %SATI. String types in Hexadecimal format (refer to AT+CSCS)

AT%SATE Send SAT Envelope Command

AT%SATE	Send SAT Envelope Command
Command Function	This command sends a SAT command to the SIM, using the envelope mechanism of SIM application toolkit.
Syntax Query	N/A
Syntax	N/A
Write Syntax	AT%SATE=<satCmd>
Write Response	%SATE: <satRsp> OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<satCmd>	String type: SIM application toolkit command, starting with command tag
<satRsp>	String type: SIM application toolkit response, starting with first bye of response data
Notes	Associated commands AT%SATT,AT%SATC, AT%SATR. Associate results codes %SATE, %SATA, %SATN and %SATI. String types in Hexadecimal format (refer to AT+CSCS)

AT%SATR Send SAT Command Response

AT%SATR	Send SAT Command Response
Command Function	This command sends a SAT response to a previously received SAT command.
Syntax Query	N/A
Syntax	N/A
Write Syntax	AT%SATR=<satRsp>
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<satRsp>	String type: SIM application toolkit response, starting with first bye of response data.
Notes	Associated commands AT%SATT,AT%SATC, AT%SATE. Associated results codes %SATE, %SATA, %SATN and %SATI. String types in Hexadecimal format (refer to AT+CSCS)

AT%SATT Terminate SAT Command or Session

AT%SATT	Terminate SAT Command or Session
Command Function	This command is used to terminate a SIM application toolkit command or session
Syntax Query	N/A
Syntax	N/A
Write Syntax	AT%SATT=<cs>
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<cs>	0 = user stop redialing 1 = end of redialing reached 2 = user ends session
Notes	Associated commands: AT%SATR, AT%SATC, AT%SATE. Associated results codes: %SATE, %SATA, %SATN and %SATI. String types in Hexadecimal format (refer to AT+CSCS)

Miscellaneous Commands

AT$AUDPLY Audio File Play

AT$AUDPLY	Audio File Play
Command Function	This command is used to play a 5.15 kbit/sec AMR audio file that is stored in FFS
Syntax Query	AT$AUDPLY=?
Syntax	$AUDPLY: “FILENAME” OK
Write Syntax	AT$AUDPLY=”<FILENAME>”
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
“<FILENAME>”	AMR audio filename
Notes	$VSELECT controls the audio path $VLVL controls speaker volume $FTPGET may be used to store AMR audio files in the FFS $FFS may be used to list the files in the FFS
Examples	

AT$AUDSTP Audio File Stop Playing

AT$AUDSTP	Audio File Stop Playing
Command Function	This command is used to stop playing an AMR audio file that is stored in FFS.
Syntax Query	AT$AUDSTP=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$AUDSTP
Execute Response	OK
Unsolicited Response	N/A
Parameter Values	N/A
Notes	N/A
Examples	N/A

AT$ICCID Integrated Circuit Card ID

AT$ICCID	Integrated Circuit Card ID
Command Function	This command allows the user to retrieve the Integrated Circuit Card Identification (ICCID) from the SIM.
Syntax Query	AT$ICCID=?
Syntax	OK
Write Syntax	N/A
Write Response	OK
Read Syntax	AT$ICCID?
Read Response	$ICCID: <iccid> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	N/A
Notes:	A SIM must be inserted into the module and initialized in order to read the ICCID of the SIM

AT%CGPCO Set Type of Authentication, Username and Password

AT%CGPCO	Set Type of Authentication, Username and Password
Command Function	This command sets the type of Authentication, username and password for GPRS context activation.
Syntax Query	AT%CGPCO=?
Syntax	%CGPCO: 0,(0-251),(0-6) OK
Write Syntax	AT%CGPCO=<Input format>,“<Authentication data>”,<cid>
Write Response	OK
Read Syntax	AT%CGPCO?
Read Response	CGPCO: 0,"<PCO Hex string>",1 CGPCO: 0,"<PCO Hex string>",2 OK AT%CGPCO? CGPCO: 1,"<Username,Password>",1 CGPCO: 1,"<Username,Password>",2 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<Input format>	0 = Inputs specified in Hexadecimal 1 = Inputs specified in ASCII

AT%CGPCO	Set Type of Authentication, Username and Password
<Authentication data>	Authentication data (ASCII) <username>,<password> where Username: Maximum 63 bytes ASCII string. Password: Maximum 63 bytes ASCII string. Authentication data (Hexadecimal): Protocol Configuration Option specified in Hex value; maximum size is equal to 251 bytes.
<cid>	0 = The new username and password is to be applied to all context Activation. 1 = The new username and password is to be applied to Context identifier 1. 2 = The new username and password is to be applied to Context identifier 2.
Notes	AT+CGDCONT command must be set before the %CGPCO command is used. If %CGPCO is set with the input format of 0 (hexadecimal), then the setting of AT%CGPPP will be ignored. Username and Password are case sensitive.
Example:	Example of ASCII input parameters: AT%CGPCO=1, “username, password”, 1 AT%CGPCO? CGPCO: 1,"username,password",1 (PAP:80C023160101001608757365726E616D65087061737 776F726480211001010010810600000000830600000000) Example of Hex input parameters: AT%CGPCO=0, “80C023160101001608757365726E616D650870617373 776F726480211001010010810600000000830600000000”, 1

AT%CGPPP Negotiation Selection

AT%CGPPP	PPP Negotiation Selection
Command Function	This command is used select the type of negotiation protocol.
Syntax Query	AT%CGPPP=?
Syntax	%CGPPP: (0-3) OK
Write Syntax	AT%CGPPP=<pt>
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<pt>	(authentication protocol) 0 = No authentication (ignore login + pwd) 1 = PAP 2 = CHAP 3 = automatic authentication
Notes	This command is used in conjunction with the %CGPCO command.

AT%CSTAT Unsolicited SIM status

AT%CSTAT	Unsolicited SIM status
Command Function	Enable/disable unsolicited status reports from SIM processes
Syntax Query	AT%CSTAT=?
Syntax	%CSTAT: (0,1)
Write Syntax	AT%CSTAT=<mode>
Write Response	OK
Read Syntax	AT%CSTAT?
Read Response	%CSTAT: <mode> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = disabled 1 = enabled
Notes	
Example:	AT%CSTAT=1 After power on, the following unsolicited results codes will be delivered to the SIM as the processes are have been initialized and are initialized. %CSTAT: EONS, 0 - EONS not ready %CSTAT: PHB, 1 - Phonebook ready %CSTAT: SMS, 1 - SMS Ready %CSTAT: RDY, 1 - All SIM functions ready

AT%EM Engineering Mode

AT%EM	Engineering Mode
Command Function	This command allows the user to view engineering mode functions including Serving cell and neighboring cell information
Syntax Query	AT%EM=?
Syntax	%EM: (2-3),(1-13) OK
Write Syntax	AT%EM=<mode>,<type>
Write Response	OK
Read Syntax	AT%EM?
Read Response	Error
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
< mode >	2= AT Command 3 =PCO
<type>	See Engineering Mode Document
Notes	Please see the Engineering Mode Manual Technical Note GSM0000TN012 for complete details of this command.

AT%MEPD MEPD Configuration Data

AT%MEPD	MEPD Configuration Data
Command Function	This command allows the user to read the current values of the MEPD configuration data. This configuration data is used for SIM personalization. The data is entered using a special application to bind the MEPD data to the modem. This AT command can only query the contents of the data, not change it.
Syntax Query	AT%MEPD=?
Syntax	%MEPD: ("MAX","ALE","RFM","RFA","RSM","RSA","TMF","ETF","AIF", "NPL","CPL","PPL") OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$MEPD=<type>
Execute Response	%MEPD: <data per “type”> OK
Unsolicited Response	N/A
Parameter Values	

AT%MEPD	MEPD Configuration Data
<type>	<ASCII string> specifying MEPD data type “MAX” - Max value of failure counter “ALE” - Current value of failure counter “RFM” - Failure counter RESET fail max value “RFA” - Failure counter RESET fail current value “RSM” - Failure counter RESET success value “RSA” - Failure counter RESET success current value “TMF” - Timer flag “ETF” - ETSI flag “AIF” - AIRTEL flag “NPL” - Network personalization lock “CPL” - Corporate personalization lock “PPL” - Provider personalization lock
Notes:	“NPL” MEPD data replaces AT+CLCKCFG. “CPL” MEPD data replaces AT+CLCKCP. “PPL” MEPD data replaces AT+CLCKSP.

AT%SIMIND SIM Status Indication

AT%SIMIND	SIM Status Indication
Command Function	Enable/disable reporting of unsolicited SIM status indications for SIM removal and SIM insertion.
Syntax Query	AT%SIMIND=?
Syntax	%SIMIND: (0,1) OK
Write Syntax	AT%SIMIND=<mode>
Write Response	OK
Read Syntax	AT%SIMIND?
Read Response	%SIMIND: <mode> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<mode>	0 = reporting disabled (default) 1 = reporting enabled

AT%SIMIND	SIM Status Indication
Notes	Unsolicited indications use the following format: “%SIMREM: 1” indicates that the SIM has been removed, and “%SIMINS: 1” indicates that the SIM has been inserted. SIM removal and insertion checks can be performed using either software or hardware methods as described under the AT$SIMDTC AT command. Software SIM detection is dependent on the current setting of the phone/SIM functionality (AT+CFUN). If phone/SIM functionality is disabled (AT+CFUN=0) then software SIM detection is disabled as well (i.e. %SIMREM, %SIMINS URCs are not generated). Software SIM detection is fully functional when AT+CFUN=1. Hardware SIM detection is partially dependent on the current setting of the phone/SIM functionality (AT+CFUN). If phone/SIM functionality is disabled (AT+CFUN=0) then hardware SIM removal detection is disabled as well (i.e. %SIMREM URC is not generated). Hardware SIM insert detection is only dependent on the setting of AT$SIMDTC. Phone/SIM functionality is performed automatically under the control of the AT$AREG command. When AT$AREG=1 or 2 the modem will automatically enable phone/SIM functionality (as well as register). However when AT$AREG=0 phone/SIM functionality is disabled and the behavior described above regarding SIM detection and URC generation applies. Another feature of SIM insert detection is that the modem will automatically register on the network if AT$AREG=1 2 or 3.

AT$SIMCFG SIM Configuration

AT$SIMCFG	SIM Configuration
Command Function	This command allows configuration of the SIM speed enhancement.
Syntax Query	AT$SIMCFG=?
Syntax	$SIMCFG: (0-1) OK
Write Syntax	AT$SIMCFG=<sim_speed>
Write Response	OK
Read Syntax	AT$SIMCFG?
Read Response	$SIMCFG: <sim_speed>
Execute Syntax Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<sim_speed>	0 – SIM speed enhancement is enabled. 1 – SIM speed enhancement is disabled.
Notes	This command should only be used for SIMs that do not support speed enhancement, but indicate to the modem that speed enhancement is supported. This command must be issued before the modem accesses the SIM initially.
Example	N/A

AT%SLEEP Select level of sleep mode

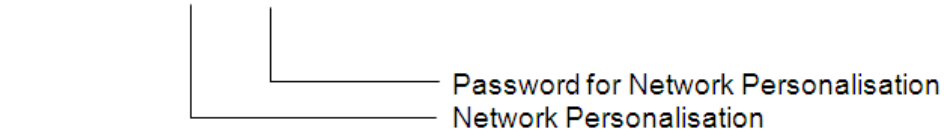
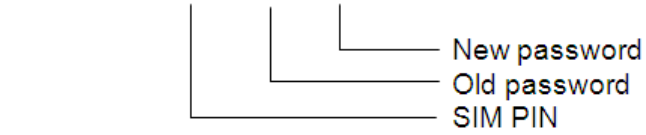
AT%SLEEP	Select level of sleep mode
Command Function	This command allows the user to select the level of sleep the modem will enter during periods of inactivity.
Syntax Query	AT%SLEEP=?
Syntax	%SLEEP: (0-4) OK
Write Syntax	AT%SLEEP=<mode>
Write Response	OK
Read Syntax	AT%SLEEP?
Read Response	%SLEEP: <mode> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	

AT%SLEEP	Select level of sleep mode
< mode >	0 = no sleep 1= Small 2 = Big 3 = Big + Deep 4 = Small+ Big +Deep No Sleep: all sections of the modem will remained powered on and ready for use. Small: All functions are active and perform normally. Some peripherals are in sleepstate. Big: The radio and peripherals are asleep. The UART is awake and able to receivedata on the serial port. The radio will wakeup periodically check for pages form thewireless Network. Deep: only the 32 MHZ clock and a small portion of the DSP are running, all othersections of the module are in a sleep state. The UART is asleep, but will wakeup with the first character received (this character will be lost). The modem willwakeup periodically to check for incoming pages form the wireless carrier.
Notes	If %SLEEP <mode> of 3 or 4 is selected, and the modem has entered Deep sleep, the UART will miss the first character that is sent over the serial port. This first character will wake up the UART and subsequent characters will be accepted by the UART. Default setting is 2

AT+CPWD Change Password

AT+CPWD	Change Password
Command Function	This command is used to set a new password for the facility lock function defined by command Facility Lock +CLCK.
Syntax Query	AT+CPWD=?
Syntax	+CPWD: (“SC”,“AD”,“OI”,“OX”,“AI”,“IR”,“AB”,“AG”,“AC”,“P2”,“PC”,“PP”,“PS”,“PN”,“-PU”,“PF”) OK
Write Syntax	AT+CPWD = <fac>,[<oldpwd>],<newpwd>
Write Response	OK or +CME ERROR: <err>
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	

AT+CPWD	Change Password
<fac>	“SC” - (SIM PIN 1) “AO” - (Barr All Outgoing Calls) “OI” - (Barr Outgoing International Calls) “OX” - (Barr Outgoing International Calls except Home Country) “AI” - (Barr All Incoming Calls) “IR” - (Barr Incoming Calls when Roaming outside the Home Country) “AB” - (All Barring Services) “AG” - (All Outgoing Barring) “AC” - (All incoming Barring) “P2” - (SIM PIN 2) "PC" - (Corporate Personalization, allows personalization to custom corporate group settings) "PP" - (Provider Personalization, allows for personalization to custom service provider defined groups) "PS" - PH-SIM (lock PHone to SIM card) (ME asks password when other than current SIM card inserted; ME may remember certain amount of previously used cards thus not requiring password when they are inserted) "PF" - lock Phone to the very First inserted SIM card (also referred in the present document as PH-FSIM) (ME asks password when other than the first SIM card is inserted) "PN" - Network Personalization (refer GSM 02.22 [33]) "PU" - network sUbset Personalization (refer GSM 02.22 [33])
<oldpwd>	Password specified for the facility. If an old password has not yet been set, <oldpwd> is not entered
<newpwd>	“new password”

AT+CPWD	Change Password
Notes	In order to change the password, the applicable facility must be enabled. See AT+CLCK. The length of the personalization password (“PC” to “PU”) is 8-16 digits. The password can be changed only if the facility is unlocked. Quotation marks are optional when entering <oldpwd> or <newpwd>.
Example:	Enter first password for Network Personalization: AT+CPWD=”PN”,,”xxxx”  To change SIM PIN: AT+CPWD=”SC”,”xxxx”,”yyyy” 

AT$MDSTAT Query Modem Status

AT$MDSTAT	Query Modem Status
Command Function	This command allows the user to display many modem status parameters with a single command.
Syntax Query	AT$MDSTAT=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	AT$MDSTAT?

AT$MDSTAT	Query Modem Status
Read Response	$DEVTYP: 0,9142 $PKG: 50 OEM_MODEM +CPIN: READY HPLMN: 310,410 +CREG: 1, CB90, 290F %CGREG: 1, CB90, 290F +COPS: 310,410 ARFCN: 198 BAND: 850 +CSQ: 14,0 +CGACT: 1,0 $NETIP: "000.000.000.000", "000.000.000.000", "000.000.000.000" DREG_NW: 0001 DREG_ME: 0005 DT_CAUSE: CAFF DT_TYPE: 0004 $LUPREJ: 0,0 $RAUREJ: 0,0 +CEER: 0,1,1,255,no error $CGEER: no PDP reject cause TCP ERROR: 0 PAD DISC: 0 API DISC: 5 OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<$DEVTYP>	see AT$DEVTYP? command
<$PKG>	see AT$PKG command
<+CPIN>	see AT+CPIN? command
<HPLMN>	Country code and Network code from SIM
<+CREG>	see AT+CREG? command
<%CGREG>	see AT%CGREG? command
<+COPS>	see AT+COPS? command (numeric only)
<+CSQ>	see AT+CSQ command
<+CGACT>	see AT+CGACT? command

AT$MDSTAT	Query Modem Status
<$NETIP>	see AT$NETIP? command
<DREG_NW>	DREG_NW values (Deregistration - Network) FFFF = not set = no service 1 = limited service
<DREG_ME>	DREG_ME values (Deregistration - ME) 0 = not set 1 = GPRS detach 2 = IMSI detach 3 = Combined detach 4 = SIM removed 5 = Power off 6 = GPRS disabled 7 = Limited service 8 = Soft power off
<DT_CAUSE>	see table in notes section below
<DT_TYPE>	DT_TYPE (Detach type) 1 = GPRS detach 2 = IMSI detach 3 = Combined detach 4 = SIM removed 5 = Power off 6 = GPRS disabled 7 = Limited service 8 = Soft power off
<$LUPREJ>	see AT$LUPREG? command
<$RAUREJ>	see AT$RAUREG? command
<+CEER>	see AT+CEER command

AT$MDSTAT	Query Modem Status
<TCP ERROR>	TCP ERROR values 0 = no error stored 1 = normal closure 2 = unexpected RST 3 = unexpected SYN 4 = received FIN 5 = timeout establishing connection 6 = failure establishing connection 7 = timeout in close 8 = timeout on established connection 9 = RST on established connection 10 = ICMP rec'd: no route to host 11 = ICMP rec'd: connection refused 12 = ICMP rec'd: frag req, DF set 13 = no route found for dest 14 = interface down or other failure 15 = internal stack failure
<PAD DISC>	0 = no reason stored 1 = server timeout 2 = PPP caused 3 = host caused 4 = connection timeout 5 = caused by AT cmd
<API DISC>	0 = no reason stored 4 = connection timeout 5 = caused by AT cmd 6 = bad msg format 7 = connect fail

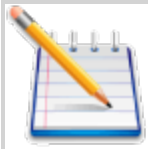
AT$MDSTAT	Query Modem Status
Notes	DT_CAUSE values (Detach cause) 0402 IMSI unknown in HLR 0403 Illegal MS 0404 IMSI unknown in VLR 0405 IMEI not accepted 0406 Illegal ME 040b PLMN not allowed 040c Location Area not allowed 040d Roaming not allowed in this location area 0411 Network failure 0416 Congestion 0420 Service option not supported 0421 Requested service option not subscribed 0422 Service option temporarily out of order 0426 Call cannot be identified 0430 Retry upon entry into a new cell (mapped 0x0430..0x043f -> 0x0430) 045f Semantically incorrect message 0460 Invalid mandatory information 0461 Message type non-existent or not implemented 0462 Message type not compatible with the protocol state 0463 Information element non-existent or not implemented 0464 Conditional IE error 0465 Message not compatible with the protocol state 046f Protocol error, unspecified c400 No error, successful operation (MM) c480 MS is not registered or deregistration started c481 Time-out in MM during establishment c482 Cell does not support call reestablishment c483 Preemptive release, e.g. MO-MT clash in MM c484 Reject, not in idle mode (MMR_PLMN_IND) c488 AUTHENTICATION REJECT received c489 A valid SIM is not present

AT$MDSTAT	Query Modem Status
Notes	c4ff No error cause (MM) 0a02 IMSI unknown in HLR 0a03 Illegal MS 0a06 Illegal ME 0a07 GPRS services not allowed 0a08 GPRS services and non-GPRS services not allowed 0a09 MS identity cannot be derived by the network 0a0a Implicitly detached 0a0b PLMN not allowed 0a0c Location Area not allowed 0a0d Roaming not allowed in this location area 0a0e GPRS services not allowed in this PLMN 0a10 MSC temporarily not reachable 0a11 Network failure 0a16 Congestion 0a30 Retry upon entry into a new cell (- 0x0A3F) 0a5f Semantically incorrect message 0a60 Invalid mandatory information 0a61 Message type non-existent or not implemented 0a62 Message type not compatible with the protocol state 0a63 Information element non-existent or not implemented 0a64 Conditional IE error 0a65 Message not compatible with the protocol state 0a6f Protocol error, unspecified 4a07 GPRS services not allowed (MS MM originated) 4a0a Implicitly detached (MS MM originated) 4a6f Protocol error, unspecified (MS MM originated) 8aff Network send message without cause information element ca00 No error, successful operation (GMM) ca80 attach_attempt counter over or equal 5 ca81 attach_attempt counter under 5 ca88 AUTHENTICATION REJECT received ca89 reserved for GACI ca8a reserved for GACI ca8b reserved for GACI, no service available ca8c reserved for GACI caff No error cause (GMM)

Accelerometer Commands

Accelerometer

If the Device requires Auto-Calibration the Accelerometer commands should only be used if issues are experienced with the accelerometer.



The optional features described in this section are not applicable to all devices. For information about which features are present for each product, please refer to the product specification sheet.

AT$ACCAC Accelerometer Auto-Calibration

AT$ACCAC	**Accelerometer Auto-Calibration**
Command Function	This command is used to set the device to auto-calibrate the x, y, and z axes for the accelerometer
Syntax Query	AT$ACCAC=?
Syntax	$ACCAC: (0-1) OK
Write Syntax	AT$ACCAC =< AUTOCAL >
Write Response	OK
Read Syntax	AT$ACCAC?
Read Response	$ ACCAC: <AUTOCAL><STATUS> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<AUTOCAL>	0 – Auto calibration off 1 – Auto calibration on

AT$ACCAC	Accelerometer Auto-Calibration
<STATUS>	0 – Initializing 1 – Getting gravity vector 2 – Gravity vector complete 3 – Getting XY vector 4 - Complete
Notes	$ACCAC cannot be set to 1 if $ACCMGC is set to 1. When calibrating the device using AT$ACCAC=1, the device must get a GPS lock before proceeding with the calibration.
Examples	

AT$ACCAM Accelerometer 'Any Motion' Commands

AT$ACCAM	Accelerometer 'Any Motion' Commands
Command Function	This command allows the user to set the 'any motion' parameters of the accelerometer. The 'any motion' setting allows users to trigger an event when the accelerometer senses motion in any axis.
Syntax Query	AT$ACCAM=?
Syntax	$ACCAM: (0-1),(0-255),(0-3),(0-64800) OK
Write Syntax	$ACCAM=<enable>,<threshold>,<count>,<holdtime>
Write Response	OK
Read Syntax	AT$ACCAM?
Read Response	$ACCAM: <enable>,<threshold>,<count>,<holdtime> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A

AT$ACCAM	Accelerometer ‘Any Motion’ Commands
Parameter Values	
<enable>	Disable/enable the any motion feature 0 = disable 1 = enable (default value)
<threshold>	Accelerometer 'any motion' threshold. This value represents the level of movement required to be considered 'any motion.' Lower values are more sensitive to motion. The <threshold> values are raw hardware units (not mili-G's). Range: 0 – 255 Default: 4
<count>	Accelerometer ‘any motion’ count Range: 0 -3 Default: 1
<holdtime>	Number of seconds of no motion before the ‘any motion’ event is cleared Range: 0 – 64800 Default: 10
Notes	In order for the accelerometer to be used to aid the GPS stability, the settings for the Any Motion feature are restricted to these values: • <enable> must be a 1 • <threshold> must be between 2 and 4 • <count> must be 1 • <hold> must be no greater than 180. If these restrictions are not met, the accelerometer will not be used for GPS stability, which can result in increased GPS position drift.
Examples	

AT$ACCCFG Accelerometer Configuration

AT$ACCCFG	Accelerometer Configuration
Command Function	This command allows the user to set or query the accelerometer configuration.
Syntax Query	AT$ACCCFG=?
Syntax	$ACCCFG: (0-2),(0-6),(0,2,3),(0-3) OK
Write Syntax	AT$ACCCFG=<Range>,<Bandwidth>,<Mode>,<Wakeup Pause>
Write Response	OK
Read Syntax	AT$ACCCFG?
Read Response	$ACCCFG: <Range>,<Bandwidth>,<Mode>,<Wakeup Pause> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<Range>	Accelerometer range: 0 = +/- 2 G (default value) 1 = +/- 4 G 2 = +/- 6 G

AT$ACCCFG	Accelerometer Configuration
<Bandwidth>	Accelerometer bandwidth: 0 = 25 Hz 1 = 50 Hz 2 = 100 Hz 3 = 190 Hz 4 = 375 Hz (default value) 5 = 750 Hz 6 = 1500 Hz
<Mode>	Accelerometer mode: 0 = Normal (default value) 2 = Sleep 3 = Wakeup
<Wakeup Pause>	Accelerometer wakeup pause 0 = 20 msec (default value) 1 = 80 msec 2 = 320 msec 3 = 2560 msec
Notes	In order for the accelerometer to be used to aid the GPS stability, the settings for the accelerometer configuration are restricted to these values: • <range> must be 2G • <bandwidth> must be between 3 and 4 (190 - 375 Hz) • <mode> must be 0 (normal) If these restrictions are not met, the accelerometer will not be used for GPS stability, which can result in increased GPS position drift.
Examples	

AT$ACCEL Query Accelerometer Reading

AT$ACCEL	Query Accelerometer Reading
Command Function	This command allows the user to read the accelerometer running average values.
Syntax Query	AT$ACCEL=?
Syntax	OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	AT$ACCEL?
Read Response	$ACCEL: <X>,<Y>,<Z>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<X>	X axis acceleration value in milliG
<Y>	Y axis acceleration value in milliG
<Z>	Z axis acceleration value in milliG
Notes	
Examples	

AT$ACCFLT Accelerometer Filter Parameters

AT$ACCFLT	Accelerometer Filter Parameters
Command Function	This command allows the user to set the accelerometer filter threshold and parameters
Syntax Query	AT$ACCFLT=?
Syntax	$ACCFLT: (1-6), (-2000 - -10, +10 - +2000),(1-25),(0-2000),(1-20) OK
Write Syntax	AT$ACCFLT=<Filter No>,<Threshold>,<Duration>,<Hysteresis>,<Coef> OK
Write Response	OK
Read Syntax	AT$ACCFLT?
Read Response	$ACCFLT: <Filter No>,<Threshold>,<Duration>,<Hysteresis>,<Coef> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<filter no>	Number of filter
<threshold>	Filter threshold (in MilliG)
<duration>	Number of samples to exceed threshold before setting filter event
<hysteresis>	Filter Hysteresis (in MilliG)
<coef>	Filter coefficient

AT$ACCFLT	Accelerometer Filter Parameters
Notes	

Defaults:	1	2	3	4	5	6
<Thresh-old>	250	-250	250	-250	12-50	750
<Dura-tion>	25	25	25	25	25	25
<Hys-teresis>	25	25	25	25	25	25
<Coef>	10	10	10	10	10	10

AT$ACCMGC Set Query Accelerometer orientation auto-calibration

AT$ACCMGC	Set Query Accelerometer orientation auto-calibration
Command Function	This command allows the user to set the accelerometer orientation auto-calibration configuration.
Syntax Query	AT$ACCMGC=?
Syntax	$ACCMGC: (0-1),(0-3) OK
Write Syntax	AT$ACCMGC=<Mode>,<Cmd>
Write Response	OK
Read Syntax	AT$ACCMGC?
Read Response	$ACCMGC: <Mode>,<Status> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	NA
Parameter Values	
<Mode>	MagiCal mode 0 - Disabled 1 - Enabled
<Cmd>	MagiCal command 0 - No command 1 - Start static calibration 2 - Start dynamic calibration 3 - Re-init the dynamic calibration

AT$ACCMGC	Set Query Accelerometer orientation auto-calibration
<Status>	MagiCal status 0 - Calibration not started 1 - Static calibration in progress 2 - Static calibration completed 3 - Dynamic calibration in progress 4 - Dynamic calibration completed
Note	AT$ACCMGC cannot be set to 1 if $ACCAC is set to 1.

AT$ACCMGC	Set Query Accelerometer orientation auto-calibration
Notes	Procedure for orientation calibration: Preparation: Mount the device in the vehicle. The device should be securely mounted so that it can't shift. If it does shift the orientation calibration must be re-done. Static orientation calibration: With the vehicle level and stationary, initiate the first step, with the AT cmd: AT$ACCMGC=1,1 This will collect data for 30 seconds. The progress of the command can be checked with: AT$ACCMGC? $ACCMGC: 1, 1 When the 30 second cal is complete, the status will read: AT$ACCMGC? $ACCMGC: 1, 2 The second param of '2' indicates that the static calibration is complete. Static orientation calibration: The next step is the 'Dynamic calibration'. To do this you need 100 ft or so of straight and level space that the vehicle can be driven in. Position the vehicle at the starting point. Initiate the dynamic cal with the AT cmd: AT$ACCMGC=1,2 This starts a two minute interval when the device monitors the acceleration. The installer should quickly and safely accelerate the vehicle in a straight line for 2-3 seconds, then brake to a halt. When the vehicle is stopped, issue this AT cmd to verify completion of the dynamic calibration: AT$ACCMGC? $ACCMGC: 1, 4 The second param of '4' indicates that the device was able to collect enough samples to complete the dynamic calibration. If the device responds: AT$ACCMGC? $ACCMGC: 1, 3 It is still waiting for the calibration acceleration threshold to be exceeded. If the device responds: AT$ACCMGC? $ACCMGC: 1, 2 The device has either timed out or did not receive enough samples and has gone back to the 'Static calibration complete' state. The dynamic cal was not successful and must be re-done. If the device is moved in the vehicle, both the static and dynamic calibration must be re-done. The dynamic orientation calibration may be re-done as needed without re-doing the static calibration. However, if the static calibration is re-done, the dynamic calibration must be re-done as well. The re-init command may only be give with the mode of 0. This command disables the auto calibration and clears the saved calibration data. This sets the auto calibration status to 0 (Calibration not started).

AT$ACCORN Device Orientation

AT$ACCORN	Device Orientation
Command Function	This command allows the user to set the device orientation with respect to the vehicle body.
Syntax Query	AT$ACCORN=?
Syntax	$ACCORN: (-90 – +90),(-180 – +180),(-180 – +180) OK
Write Syntax	AT$ACCORN=<Pitch>,<Roll>,<Yaw>
Write Response	OK
Read Syntax	AT$ACCORN?
Read Response	$ACCORN: <Pitch>,<Roll>,<Yaw> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<Pitch>	Device angle about Y axis relative to the vehicle Default: 0
<Roll>	Device angle about X axis relative to the vehicle Default: 0
<Yaw>	Device angle about Z axis relative to the vehicle Default: 0
Notes	The orientation parameters are used to convert the accelerometer axes to the vehicle axes so that acceleration and deceleration along the vehicle centerline (front-to-back) can be measured separately from the side-to-side and up-and-down accelerations.

AT$ACCSAM Set/Query Accelerometer Sampling

AT$ACCSAM	Set/Query Accelerometer Sampling
Command Function	This command allows the user to set the accelerometer sampling rate and filter coefficient.
Syntax Query	AT$ACCSAM=?
Syntax	$ACCSAM: (0-25),(1-20)
Write Syntax	AT$ACCSAM=<sample rate>,<coef>
Write Response	OK
Read Syntax	AT$ACCSAM?
Read Response	AT$ACCSAM: <sample rate>,<coef> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<sample rate>	Number of samples per second to read from the accelerometer. Range: 0-25 Default: 25
<coef>	Filter coefficient Range: 1-20 Default: 10
Notes	
Examples	

AT$AUTOCFG Accelerometer Auto-Calibration Configuration

AT$AUTOCFG	Accelerometer Auto-Calibration Configuration
Command Function	This command is used to set configuration parameters that are used in the auto calibration procedure for the accelerometer.
Syntax Query	AT$AUTOCFG=?
Syntax	$AUTOCFG: (5-30), (10-60), (100-1000), (10-250), (0-1), (5-40), (5-50) OK
Write Syntax	AT$ AUTOCFG =< Z Cal Time >,< MinSamples >,<Thresh>,< Hysteresis >, < Direction >,<Speed Thresh>, <Lat Acc Thresh>
Write Response	OK
Read Syntax	AT$ AUTOCFG?
Read Response	$ AUTOCFG: <THRHLD><SET TIME><CLR TIME> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<Z Cal Time>	Time in seconds used to calibrate the Z-Axis while not moving
<MinSamples>	Minimum number of samples required for Dynamic calibration
<Thresh>	Minimum number of samples required for Dynamic calibration
<Hysteresis>	Hysteresis for G threshold for Dynamic calibration (mG)

AT$AUTOCFG	Accelerometer Auto-Calibration Configuration
<Direction>	Direction for Dynamic calibration 0 - acceleration (positive acceleration) of device used for dynamic calibration 1 - braking (negative acceleration) of device used for dynamic calibration
<Speed Thresh>	Minimum Speed threshold for Dynamic calibration (units of knots)
<Lat AccThresh>	Lateral G threshold for Dynamic calibration (units of mG)
Notes	
Examples	

AT$MGCCFG Accelerometer MagiCal Configuration

AT$MGCCFG	Accelerometer MagiCal Configuration
Command Function	This command allows the user to set or query the accelerometer orientation auto-calibration configuration.
Syntax Query	AT$MGCCFG=?
Syntax	$MGCCFG: (30-600),(10-60),(100-1000),(10-250),(0-1) OK
Write Syntax	AT$MGCCFG=<Timeout>,<MinSamples>,<Thresh>,<Hysteresis>, <Direction>
Write Response	OK
Read Syntax	AT$MGCCFG?
Read Response	$MGCCFG: <Timeout>,<MinSamples>,<Thresh>,<Hysteresis>, <Direction> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<Timeout>	Seconds for Dynamic calibration timeout Range: 30-600 Default: 120
<MinSamples>	Minimum number of samples required for Dynamic calibration Range: 10-60 Default: 20
<Thresh>	G threshold for Dynamic calibration (units of mG) Range: 100-1000 Default: 250

AT$MGCCFG	Accelerometer MagiCal Configuration
<Hysteresis>	Hysteresis for G threshold for Dynamic calibration (mG) Range: 10-250 Default: 20
<Direction>	Direction for Dynamic calibration 0 = acceleration (positive acceleration) of device used for dynamic calibration 1 = braking (negative acceleration) of device used for dynamic calibration Default: 0
Notes	

GPS Commands

AT$ALTOSI Set and Query GPS Overspeed Interval

AT$ALTOSI	Set and Query GPS Overspeed Interval
Command Function	This command allows the user to define the criteria for a GPS overspeed event. A GPS overspeed event occurs when the minimum speed that is defined by the <speed> parameter is maintained for a specific duration of time.
Syntax Query	AT$ALTOSI=?
Syntax	$ALTOSI: (0 – 65535),(0-65535),(0-1) OK
Write Syntax	AT$ALTOSI=<speed>,<interval>,<sticky>
Write Response	OK
Read Syntax	AT$ALTOSI?
Read Response	$ALTOSI: <speed>,<interval>,<status>,<max_speed>,<duration> OK
Execute Syntax	AT$ALTOSI
Execute Response	ERROR
Unsolicited Response	N/A
Parameter Values	
<speed>	Speed, in knots, must be met and/or exceeded to trigger the GPS overspeed event.
<interval>	Number of consecutive seconds for which <speed> must be maintained to trigger the GPS overspeed event.
<sticky>	Optional parameter to make overspeed parameters persistent. If 1, then speed/interval are written to flash immediately (no AT&W required). Default = 0.

AT$ALTOSI	Set and Query GPS Overspeed Interval
<status>	If 1, then <max_speed> and <duration> represent a GPS overspeed interval that is currently active. If 0, they represent the previous GPS overspeed interval.
<max_speed>	The highest speed (in knots) that was attained in the current or previous GPS overspeed interval.
<duration>	Number of consecutive seconds that the speed was at or above <speed>.
Notes	If <speed> is set to zero, the GPS overspeed event is disabled.

AT$EXCACC Excessive Acceleration

AT$EXCACC	Excessive Acceleration
Command Function	This command is used to configure the excessive acceleration event. Input event 198
Syntax Query	AT$EXCACC=?
Syntax	$EXCACC: (0-200),(1-20),(1-20) OK
Write Syntax	AT$EXCACC=<THRESH>,<SET TIME >,<CLR TIME >
Write Response	OK
Read Syntax	AT$EXCACC?
Read Response	$EXCACC: <THRESH>,<SET TIME >,<CLR TIME> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<THRESH>	0 = Excessive Acceleration Event Disabled 1-200 = Acceleration Threshold (in Knots * 10)
<SET TIME>	1-20 = Number of seconds above threshold to set event

AT$EXCACC	Excessive Acceleration
<CLR TIME>	1-20 = Number of seconds below threshold to clear event
Notes	A value of 100 for threshold equals 10 knots per second. This function uses GPS to determine acceleration.

AT$EXCDEC Excessive Deceleration

AT$EXCDEC	Excessive Deceleration
Command Function	This command is used to configure the excessive deceleration event. Input event 199.
Syntax Query	AT$EXCDEC=?
Syntax	$EXCDEC: (0-200),(1-20),(1-20) OK
Write Syntax	AT$EXCDEC=<THRESH>,<SET TIME >,<CLR TIME >
Write Response	OK
Read Syntax	AT$EXCDEC?
Read Response	$EXCDEC: <THRESH>,<SET TIME >,<CLR TIME> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<THRESH>	0 – Excessive Deceleration Event Disabled 1-200 – Deceleration Threshold (in Knots * 10)
<SET TIME>	1-20 – Number of seconds above threshold to set event
<CLR TIME>	1-20 – Number of seconds below threshold to clear event
Notes	A value of 100 for threshold equals 10 knots per second. This function uses GPS to determine deceleration.

AT$GEOFNC Geo Fencing a Circular Area

AT$GEOFNC	Geo Fencing a Circular Area
Command Function	This command allows a user to send a GPS message when the device moves in or out of a geographical area.
Syntax Query	AT$GEOFNC=?
Syntax	$GEOFNC: (1-25),(0-1000000),(-90.0 - +90.0),(-180.0 - +180.0) OK
Write Syntax	AT$GEOFNC=<fenceNum> <radius>,<latitude>,<longitude>
Write Response	OK
Read Syntax	AT$GEOFNC?
Read Response	$GEOFNC: <fenceNum>,<radius>,<latitude>,<longitude> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<fenceNum>	Defines the fence number
<radius>	Defines radius of the circle from given Latitude and Longitude coordinates (in meters)
<latitude>	Defines the latitude for the center point of a circle
<longitude>	Defines the longitude for the center point of a circle

AT$GEOFNC	Geo Fencing a Circular Area
Notes	An AT$EVENT command has to be set to send a GPS message to the remote host when entering or exiting the fenced area. See GSM2000CB001 – Mobile Tracker Event Cookbook to see an example. Although this command accepts latitude/longitude parameters with up to 15 characters, internally the value is stored as a C float type which has less precision (but requires half the storage size). The float type is capable of storing accuracy commensurate with the GPS receiver's capability, but the queried latitude/longitude values of the AT$GEOFNC command may differ from the input parameters due to this precision limitation.

AT$GFDBNC Set Geofence Debounce Count

AT$GFDBNC	Set Geofence Debounce Count
Command Function	This command allows the user to set the number of consecutive geofence positions required to trigger an 'inside geofence' or 'outside geofence' event. It prevents a false reading when the device is on the threshold of the geofence.
Syntax Query	AT$GFDBNC=?
Syntax	$GFDBNC: (0-250),(0-250) OK
Write Syntax	AT$GFDBNC=<out_cnt>,<in_cnt>
Write Response	OK
Read Syntax	AT$GFDBNC?
Read Response	$GFDBNC: <out_cnt>, <in_cnt> OK
Execute Syntax	N/A
Execute Response	
Unsolicited Response	N/A
Parameter Values	
<out_cnt>	Consecutive GPS position reports outside a geofence required to trigger '0' condition for geofence input event (see $EVENT)
<in_cnt>	Consecutive GPS position reports inside a geofence required to trigger '1' condition for geofence input event (see $EVENT)
Notes	The GPS reporting interval varies depending on the product. for the MT-Gx and MTxxxx products, the updates are sent every one second.

AT$GFDEL Delete a Range of Geo-Fences

AT$GFDEL	Delete a Range of Geo-Fences
Command Function	This command deletes a range of geo-fences.
Syntax Query	AT$GFDEL=?
Syntax	$GFDEL: (1-25),(1-25) OK
Write Syntax	AT$GFDEL=<start>,<stop>
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<start>	First geo-fence index in range to be deleted
<stop>	Last geo-fence index in range to be deleted.
Notes	N/A

AT$GFIDX# Geo Fencing Index

AT$GFIDX#	Geo Fencing Index
Command Function	This command allows the user to query the modem for a single geofence from storage. The modem returns the index, radius, longitude and latitude. Longitude and latitude data is formatted for use in the GPS. Replace the # with the index number noted below.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	AT$GFIDX<index>?
Read Response	
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	NA
Parameter Values	
<index>	(1-25) - Index to the geofence stored in the modem
Notes	N/A

AT$GPSCMD GPS Command

AT$GPSCMD	GPS Command
Command Function	This command allows the user to configure the state of the GPS module.
Syntax Query	AT$GPSCMD=?
Syntax	$GPSCMD: (0-3) OK
Write Syntax	AT$GPSCMD=<cmd>
Write Response	OK
Read Syntax	AT$GPSCMD?
Read Response	$GPSCMD: <cmd> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<cmd>	0 – Disables the GPS 1 – Cold start 2 – Hot start 3 – Stop the current GPS fix
Notes	0 - GPS is in an idle state 1 - Commands GPS to begin acquisition from a cold start. 2 - Hot start can only be accomplished if it has valid time, ephemeris, almanac and position data. 3 - Some Enfora products do not tracking GPS, and is defined as GPS fix stop (for power consumption) After a power up or reset, the modem will attempt an assisted cold start. This is defined as the modem supplying the GPS receiver any valid time, ephemeris, almanac, and position data that the modem might have stored.

AT$GPSFD Restore GPS Filter Defaults

AT$GPSFD	Restore GPS Filter Defaults
Command Function	This command restores GPS filter defaults for the $GPSQUAL, $GPSFLT, and $GPSRST commands without impacting the rest of the system configuration. This command is intended to be used after a software upgrade to take advantage of new filter settings without having to perform a full AT&F. This command accepts one parameter (a 'sticky' flag). If the sticky flag is set, the GPS filter defaults will be persistent. If the sticky flag is not set, the GPS filter defaults will not be maintained past the next modem reset.
Syntax Query	AT$GPSFD=?
Syntax	$GPSFD: (0-1) OK
Write Syntax	AT$GPSFD=<sticky>
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<sticky>	0 = The GPS modem defaults will not be maintained past the next modem reset. 1 = The GPS modem defaults will be persistent.
Notes	

AT$GPSFLT GPS Speed Filter

AT$GPSFLT	GPS Speed Filter
Command Function	At times the GPS chipset used by this device will report non-zero speeds when the device is not moving. Normally, the positions are accurate but the false speeds cause errors in the generation of the GPS idle, distance and odometer features. This filter is usually able to detect these false motions and clamp the reported speed to 0.0 knots while still reporting valid position data. Under normal circumstances, the default values should provide acceptable GPS performance for most applications. Please contact Enfora customer support if you feel the filter settings need to be changed for your application.
Syntax Query	AT$GPSFLT=?
Syntax	If the $GPSFLT debug flag is not set for expanded AT command response, the response will be as follows: $GPSFLT: (0-2) OK If the $GPSFLT debug flag is set for expanded AT command response, the response will be as follows: $GPSFLT: md,tr,hdgD,tiAc,n4aAc,tiHdg,n4aHdg,pDop,dbg,unc,mot,ignB,tAdj OK
Write Syntax	AT$GPSFLT= <md>,<tr>,<hdgD>,<tiAc>,<n4aAc>,<tiHdg>, <n4aHdg>,<pDop>,<dbg>,<unc>,<mot>,<ignB>,<tAdj>
Write Response	OK
Read Syntax	AT$GPSFLT?

AT$GPSFLT	GPS Speed Filter
Read Response	AT$GPSFLT: <md>,<tr>,<hdgD>,<tiAc>,<n4aAc>,<tiHdg>, <n4aHdg>,<pDop>,<dbg>,<unc>,<mot>,<ignB>,<tAdj> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<md>	0 = filter is disabled 1 = filter is enabled 2 = automatic (default). Filter determines whether device is in parked mode or driving mode by determining the number of seconds the device has been idle. If in parked mode, filter is enabled. If in driving mode, filter is disabled. This improves the performance of the GPS idle, distance, and odometer features when coming off stop signs and red lights, while still stopping most of the false speed-related events when the device is parked.
<tr>	0-255 = Trust threshold. With each GPS report, this filter uses the other filter settings to assess a confidence level to determine if indicated motion seems valid. This confidence level is either incremented or reset with each report. The confidence level is the converted into a trust factor which is compared against this threshold value. If the trust factor is less than this threshold and the device is in the parked motion state, the speed is clamped to 0.0 knots. If the trust factor is above or equal to this threshold, the speed from the $GPRMC is allowed through and the device transitions into the driving state. Default is 16.

AT$GPSFLT	GPS Speed Filter
<hdgD>	0-180 = Heading delta threshold. This filter calculates the heading from the two most recent position reports and compares it to the heading provided in the $GPRMC sentence. If the delta between the two headings is greater than this threshold (in degrees), the confidence level is reset to zero. Default is 30 degrees.
<tiAc>	0-255 = TI acceleration threshold. This filter calculates the velocity change from the reported $GPRMC velocity in the two most recent position reports. If the reported velocity change is greater than this threshold (in knots/sec), the confidence level is reset to zero. Default is 10 knots/second.
<n4aAc>	0-255 = Enfora acceleration threshold. This filter calculates the velocity change from the distance between the two most recent position locations and based on the time delta. If the calculated velocity change is greater than this threshold (in knots/sec), the confidence level is reset to zero. Default is 10 knots/second.

AT$GPSFLT	GPS Speed Filter
<tiHdg>	0-180 = TI heading change threshold. This filter calculates the heading change from the reported $GPRMC heading in the two most recent position reports. If the heading change is greater than this threshold (in degrees/second), the confidence level is reset to zero. Default is 30 degrees/second.
<n4aHdg>	0-180 = Enfora heading change threshold. This filter calculates the heading using the two most recent position locations and based on the time delta. If the difference between the two most-recent heading calculations is greater than this threshold (in degrees/second), the confidence level is reset to zero. Default is 30 degrees/second.
<pDop>	Obsolete. This value is no longer used. The field is maintained for backwards compatibility.

AT$GPSFLT	GPS Speed Filter
<dbg>	Bit mask made up of the following bit definitions: 1 = Display all filter actions to serial port. If $GPSQUAL or $GPSFLT filters take any action the change or invalidate the GPS report, setting this bit will allow the code to display the action that was taken. 2 = Display runtime filter info to serial port. Allows code to display calculations/variables used by the filter. 4 = Display geofence timing debug info. 8 = System test flag. 16 = GPS log flag. Enables code to log raw GPS data to a file in flash for post-mortem analysis for certain filtering events. 32 = GPS message flag. Enabled code to send a message over the air and to the serial port, indicating that an abnormal GPS event has been detected and logged. 64 = Raw NMEA flag. Display raw NMEA input on the serial port. 128 = Expand $GPSFLT=? and $GPSQUAL=? Responses to provide a verbose explanation of each parameter.

AT$GPSFLT	GPS Speed Filter
<unc>	0-255 = Position uncertainty threshold. If internal position uncertainty value is greater than this value, the confidence level is reset to zero. Default is 75.
<mot>	0-999 = Motion transition threshold (in seconds). If device has been idle (no velocity) for this number of seconds, transition from driving mode to parked mode. Default is 120 seconds.
<ignB>	0-255 = Ignition bias. For devices that are connected to the ignition via the white wire, the code can detect when the ignition is turned on/off and adjust the trust threshold accordingly. (Vehicles with the ignition off would only be moving if being towed. Vehicles with the ignition on will probably be driving if not moving already.) For devices that are not connected to the ignition, this value should be set to 0. Default is 6.
<tAdj>	0 = Do not adjust RTC time if RTC time disagrees with GPS time. 1 = Adjust RTC time if RTC time and GPS time differ by more than 5 seconds. Default is 1.

AT$GPSFLT	GPS Speed Filter
Notes	There have also been cases where slow Time To First Fix (TTFF) reports have been attributed to discrepancies between the RTC time and the GPS time. When GPS filtering is enabled, the code ensures the RTC time is always with +/-5 seconds of the GPS time. This time sync feature is disabled by setting $GPSFLT=0. If GPS filtering is modified or disabled, performance degradation may result. With the implementation of this filter, the $ODOCFG command is now obsolete. For backwards compatibility, the $ODOCFG command will still be accepted, but the code will take no action on its parameters.

AT$GPSLCL - Configure Sending of GPS Message to the Serial Port

AT$GPSLCL	Configure Sending of GPS Message to the Serial Port
Command Function	This command allows the user to configure sending of GPS data on the serial port.
Syntax Query	AT$GPSLCL=?
Syntax	$GPSLCL: (0–1),(0-127) OK
Write Syntax	AT$GPSLCL=<option>,<nmeaMsgs>
Write Response	OK
Read Syntax	AT$GPSLCL?
Read Response	$GPSLCL: <option>,<nmeaMsgs> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<option>	0 – Disable sending of GPS data to the local port when the device is in AT command mode (Default) 1 – Enable sending of GPS NMEA ASCII data to the local port when the device is in AT command mode

<table>
<tr><th>AT$GPSLCL</th><th>Configure Sending of GPS Message to the Serial Port</th></tr>
<tr><td><nmeaMsgs></td><td>This field is the bit-wise OR of the type of messages desired. The user has following message options to select from.
Decimal Format
<table><tr><td>User Selectable</td><td>Type of NMEA Message</td></tr><tr><td>1</td><td>GGA</td></tr><tr><td>4</td><td>GSA</td></tr><tr><td>8</td><td>GSV</td></tr><tr><td>16</td><td>RMC</td></tr><tr><td>64</td><td>PENFG</td></tr></table></td></tr>
<tr><td>Notes</td><td>is used for debug purpose as directed by Enfora Technical Support Personnel. It is not a standard NMEA message.</td></tr>
</table>

AT$GPSLNA Enable/Disable Internal LNA

AT$GPSLNA	Enable/Disable Internal LNA
Command Function	This command is used to enable and disable the internal Low Noise Amplifier (LNA)
Syntax Query	$GPSLNA=?
Syntax	$GPSLNA: (0-2) OK
Write Syntax	$GPSLNA=<cfg>
Write Response	OK
Read Syntax	$GPSLNA?
Read Response	$GPSLNA: =<cfg>,<actual> OK
Execute Syntax	N/A

AT$GPSLNA	Enable/Disable Internal LNA
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<cfg>	2 = Automatic (default) 0 and 1 are for debug purposes only. 0 = Internal LNA enabled 1 = Internal LNA disabled
<actual>	0 = Internal LNA enabled 1 = internal LNA disabled
Notes	

AT$GPSOSI Set and Query GPS Overspeed Interval

AT$GPSOSI	Set and Query GPS Overspeed Interval
Command Function	This command allows the user to define the criteria for a GPS overspeed event. A GPS overspeed event occurs when the minimum speed that is defined by the <speed> parameter is maintained for a specific duration of time.
Syntax Query	AT$GPSOSI=?
Syntax	$GPSOSI: (0-255),(0-65536),(0-1) OK
Write Syntax	AT$GPSOSI=<speed>,<interval>,<sticky>
Write Response	OK
Read Syntax	AT$GPSOSI?
Read Response	$GPSOSI: <speed>, <interval>, <status>, <max_speed>, <duration> OK
Execute Syntax	AT$GPSOSI
Execute Response	ERROR
Unsolicited Response	N/A
Parameter Values	
<speed>	Speed, in knots, must be met and/or exceeded to trigger the GPS overspeed event.
<interval>	Number of consecutive seconds for which <speed> must be maintained to trigger the GPS overspeed event.
<sticky>	Optional parameter to make overspeed parameters persistent. If 1, then speed/interval are written to flash immediately (no AT&W required). Default = 0.
<status>	If 1, then <max_speed> and <duration> represent a GPS overspeed interval that is currently active. If 0, they represent the previous GPS overspeed interval.

AT$GPSOSI	Set and Query GPS Overspeed Interval
<max_speed>	The highest speed (in knots) that was attained in the current or previous GPS overspeed interval.
<duration>	Number of consecutive seconds that the speed was at or above <speed>.
Notes	If <speed> is set to zero, the GPS overspeed event is disabled.

AT$GPSQUAL GPS Quality Filters

AT$GPSQUAL	GPS Quality Filters
Command Function	This command allows the user to define extra criteria for the event engine to use before it reports a position fix as valid.
Syntax Query	AT$GPSQUAL=?
Syntax	If the $GPSFLT debug flag is not set for expanded AT command response, the response will be as follows: $GPSQUAL: (0-255),(0-255),(0-30) OK
Write Syntax	AT$GPSQUAL=<flg>,<hdop>,<sats>
Write Response	OK
Read Syntax	AT$GPSQUAL?
Read Response	$GPSQUAL:<flg>,<hdop>,<sats>
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	

AT$GPSQUAL	GPS Quality Filters
<flg>	Bit mask made up of the following bit definitions: 1 = Mark GPS report invalid if $GPGSA fix type is 2 (2D fix). This bit is provided for backwards compatibility with legacy products. It has not been shown to improve performance with this product. 2 = Mark GPS report invalid if the report is based on a propagated fix. 4 = Request POSITION EXTENDED report (in place of the basic POSITION report) from GPS chipset. This report includes additional info that the basic POSITION report does not. 8 = Request MEASUREMENT EXTENDED report. This report includes information about satellite speeds which can be used in filtering. Default is 12 (request both POSITION EXTENDED and MEASUREMENT EXTENDED reports).
<hdop>	0 = Ignore HDOP when filtering. (default) 1-255 = Mark GPS report invalid if HDOP value from $GPGSA sentence is less than or equal to this indicated HDOP limit. This field is provided for backwards compatibility with legacy products. It has not been shown to improve performance with this product.
<sats>	0 (default) = Disregard # of satellites during filtering. 1-30 = Mark GPS report invalid if the # of satellites in reported solution (from $GPGGA sentence) is below this value.
Notes	The units of the uncertainty threshold should not be taken literally. For example, a position uncertainty threshold of 100 meters does not guarantee that the actual location with be within 100 meters of the actual position. Please consult with Enfora technical support personnel before changing uncertainty thresholds.
Example:	

AT$GPSRD Read Current GPS ASCII data

AT$GPSRD	Read Current GPS ASCII data
Command Function	This command allows a user to read current NMEA format GPS data.
Syntax Query	AT$GPSRD=?
Syntax	$GPSRD: [(0-7F),(0-127)],(0-1) OK
Write Syntax	N/A
Write Response	N/A
Read Syntax	AT$GPSRD=<nmeaMsgs>,<decimal>
Read Response	"$GPG……." OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	NA
Parameter Values	The output NMEA sentence depends on whether the <nmeaMsgs> parameter is entered in Hex or Decimal format. By default, the <decimal> parameter is not required and <nmeaMsgs> parameter has to be entered as HEX value without the preceding "0x" characters as outlined in Hex Format table below.

<table>
<tr><th>AT$GPSRD</th><th>Read Current GPS ASCII data</th></tr>
<tr><td><nmeaMsgs></td><td>This field is the sum of the type of NMEA messages desired. A user has the following message options to select from. Maximum value for <nmeaMsgs> in this case would be 7F in Hex format or 127 in decimal format.

Hex Format

<table>
<tr><td>User Selectable</td><td>Type of NMEA Message</td></tr>
<tr><td>0x01</td><td>GGA</td></tr>
<tr><td>0x04</td><td>GSA</td></tr>
<tr><td>0x08</td><td>GSV</td></tr>
<tr><td>0x10</td><td>RMC</td></tr>
<tr><td>0x40</td><td>PENFG</td></tr>
</table>

Decimal Format

<table>
<tr><td>User Selectable</td><td>Type of NMEA Message</td></tr>
<tr><td>1</td><td>GGA</td></tr>
<tr><td>4</td><td>GSA</td></tr>
<tr><td>8</td><td>GSV</td></tr>
<tr><td>16</td><td>RMC</td></tr>
<tr><td>64</td><td>PENFG</td></tr>
</table>
</td></tr>
<tr><td><decimal></td><td>1 = <nmeaMsg> value has to be sum of User Selectable values from decimal table format

0 = select values out of hex table format</td></tr>
<tr><td>Notes</td><td>The $PENFG (Proprietary ENFora Gps) is used for debug purpose as directed by Enfora Technical Support Personnel. It is not a standard NMEA message.</td></tr>
</table>

AT$GPSRST Set and Query GPS Reset Interval

AT$GPSRST	Set and Query GPS Reset Interval
Command Function	This command allows the user to specify the duration of a continuous GPS no lock condition after which the GPS chip will be reset. The GPS will often take several minutes to get a lock when power has been cycled to the device. However, if valid RTC time is available, then the lock should be acquired more quickly. The second parameter allows the user to cut down the time without a lock before resetting the GPS chip when outside of the cold start window.
Syntax Query	AT$GPSRST=?
Syntax	$GPSRST: (0-255),(0-255),(0-255),(0-255) OK
Write Syntax	AT$GPSRST=<coldStartInterval>,<warmStartInterval> <filtCnt>
Write Response	OK
Read Syntax	AT$GPSRST?
Read Response	$GPSRST: <coldStartInterval>,<warmStartInterval> <filtCnt> OK
Execute Syntax	AT$GPSRST
Execute Response	ERROR
Unsolicited Response	N/A
Parameter Values	
<coldStartInterval>	When RTC time is not valid (year < 2010), this is the time (in tens of minutes) of GPS outage after which to reset the GPS chip and resume positioning.

AT$GPSRST	Set and Query GPS Reset Interval
<warmStartInterval>	When RTC time is valid (year >= 2010) and this value is zero, then coldStartInterval is used as defined above. When RTC time is valid (year >= 2010) and this value is greater than zero, this is the time (in minutes) of GPS outage after which to reset the GPS chip and resume positioning.
Notes	If <coldStartInterval> is set to 0, the GPS chip will not be reset after a GPS outage of any duration.
Example	AT$GPSRST=1,2 Code will allow 10 minutes to acquire a GPS lock before resetting the GPS chip following a power cycle (RTC time not valid). Once a lock has been acquired (RTC time automatically set when a lock has been acquired), the code will allow two minutes of 'no GPS lock' time before resetting the GPS chip. Since RTC time is maintained through an ignition reset, the two minute limit would be in effect following an ignition reset.

AT$GPSSAV Save GPS Data

AT$GPSSAV	Save GPS Data
Command Function	This command forces the device to save the GPS data to its FFS. This information will be used on the next power cycle as the last known GPS location.
Syntax Query	N/A
Syntax	N/A
Write Syntax	N/A
Write Response	N/A
Read Syntax	N/A
Read Response	N/A
Execute Syntax	AT$GPSSAV
Execute Response	OK
Unsolicited Response	NA
Parameter Values	N/A
Notes	

AT$GPSVAL GPS Speed Validation

AT$GPSVAL	GPS Speed Validation
Command Function	This command is used to turn GPS speed filtering on and off.
Syntax Query	AT$GPSVAL=?
Syntax	$ GPSVAL: (0-1) OK
Write Syntax	AT$ GPSVAL =< MODE ><ACC MOTION>
Write Response	OK
Read Syntax	AT$ GPSVAL?

AT$GPSVAL	GPS Speed Validation
Read Response	$ GPSVAL: < MODE >< ACC MOTION > OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<MODE>	0 – Use default for GPS Speed filter 1 – Disable GPS Speed filter
<ACC MOTION>	0 – Use accelerometer to determine motion 1 – Do not use accelerometer to determine motion This parameter is not available to all MT Products. This field is to enable the use of the accelerometer to assist the GPS filter in determining motion. This value is not relevant if <MODE> = 1.
Notes	The GPS speed filter here is the same as the GPS speed filter in $GPSFLT.
Examples	

AT$TODOM Virtual Trip Odometer

AT$TODOM	Virtual Trip Odometer
Command Function	The $TODOM command records how far the vehicle has traveled. This is identical to the $ODOMETER feature, but allows the user to reset this trip odometer without resetting the other odometer.
Syntax Query	AT$TODOM=?
Syntax	$TODOM:(0-4000000000) OK
Write Syntax	AT$TODOM=1234 (where 1234 is distance in meters)
Write Response	OK
Read Syntax	AT$TODOM?

AT$TODOM	Virtual Trip Odometer
Read Response	$TODOM: xxxx (xxxx=distance traveled in meters)
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	N/A
Notes	The user is able to set a seed value for the Trip Odometer starting at 0 but not higher than the maximum value of 4000000000. The AT&F command will not reset the odometer value to 0. (see example below) The Trip Odometer reading would be a 4-byte value starting from 0 to 4000000000 (maximum of approximately 2500000 miles before it rolls over to 0) The unit for Trip Odometer shall be in meters. The Trip Odometer history shall be updated every second. The Trip Odometer history shall be saved once a minute in modem's memory. This value shall be retained through an internal or external reset and can be read upon the next power up or during run time mode. The delta distance traveled between the minute marks could be lost due to an unexpected reset. However, the total distance traveled till the prior minute would still be preserved. The $ODOMETER value can trigger the event engine via input event 70. The $TODOM value can trigger the event engine via input event 91.
Example	Reset Trip Odometer to 0: $TODOM=0

AT$ODOMETER GPS Trip Odometer

AT$ODOMETER	GPS Trip Odometer
Command Function	The $ODOMETER command records how far the vehicle has traveled total, or in one trip. The user can reset the odometer at the beginning of a new trip. This is identical to the $TODOM feature, but allows the user to reset this odometer without resetting the other odometer.
Syntax Query	AT$ODOMETER=?
Syntax	$ODOMETER: (0-4000000000)
Write Syntax	AT$ODOMETER=1234 (where 1234 is distance in meters)
Write Response	OK
Read Syntax	AT$ODOMETER?
Read Response	$ODOMETER xxxx (xxxx=distance traveled in meters)
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	N/A
Notes	The user is able to set a seed value for the Trip Odometer starting at 0 but not higher than the maximum value of 4000000000. The AT&F command will not reset the odometer value to 0. The Virtual Odometer reading would be a 4-byte value starting from 0 to 4000000000 (maximum of approximately 2500000 miles before it rolls over to 0) The unit for Virtual Odometer shall be in METERS. The Virtual Odometer history shall be updated every second The Virtual Odometer history shall be saved once a minute in modem's memory. This value shall be retained through an internal or external reset and can be read upon the next power up or during run time mode. The delta distance traveled between the minute marks could be lost due to an unexpected external or non-modem originated reset. However, the total distance traveled till the prior minute would still be preserved.

AT$PLYCLR Clear Polygonal Geofence ?

AT$PLYCLR	Clear Polygonal Geofence
Command Function	This command allows a user to delete all the points for the indicated polygonal geofence (see $PLYFN#).
Syntax Query	AT$PLYCLR=?
Syntax	$PLYCLR: (0-24) OK
Write Syntax	AT$PLYCLR=<id>
Write Response	OK
Read Syntax	N/A
Read Response	N/A
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<id>	0-24: identifier of geographical polygon (see $PLYFN#)
Notes	

AT$PLYFN# Geo Fencing a Polygonal Area ?

AT$PLYFN#	Geo Fencing a Polygonal Area
Command Function	This command allows a user to define up to 25 separate polygonal geographical areas (# can be any number from 0 to 24). The event engine (see AT$EVENT) can be set up to send a message when the device moves in or out of the geographical area defined by the polygon. Each command defines one of 26 possible vertices or corners. The line segments of the polygon are generated by connecting the non-zero vertices in sequence (for example, point 0 is connected to point 1, which is connected to point 2, etc.). The polygon can be generated in either a clockwise or counterclockwise sequence (see Notes section below for additional details). You can specify your polygon with less than 26 vertices, but the first and last points of the polygon still must be identical The query returns all non-zero locations defined for the polygon as well as all (0,0) vertex locations that serve as a polygon terminator.
Syntax Query	AT$PLYFN#=?
Syntax	$PLYFN#: (0-25),(-90.0 - +90.0),(-180.0 - +180.0) OK
Write Syntax	AT$PLYFN#=<idx>,<latitude>,<longitude>
Write Response	OK
Read Syntax	AT$PLYFN#?
Read Response	$PLYFN#: PtIdx Latitude Longitude <idx> <latitude> <longitude> OK
Execute Syntax	N/A
Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	

AT$PLYFN#	Geo Fencing a Polygonal Area
<idx>	0-25: index for this vertex of the polygon
<latitude>	Defines the latitude for this vertex of the polygon in decimal degrees (for example, 32 degrees 30 minutes 0 seconds would be represented as 32.5 degrees since 30 minutes is exactly 1/2 of a degree). The latitude must be specified in 15 characters (including minus sign and decimal point) or less. This is the same representation for latitude as used in the AT$GEOFNC command.
<longitude>	Defines the longitude for this vertex of the polygon in decimal degrees (for example, -96 degrees 45 minutes 0 seconds would be represented as -96.75 degrees since 45 minutes is exactly 3/4 of a degree). The longitude must be specified in 15 characters (including minus sign and decimal point) or less. This is the same representation for longitude as used in the AT$GEOFNC command.
Notes	In general, it is always a good idea to verify the integrity of your polygon description with a mapping tool such as Google Earth. The most common mistake with polygons is to mistype one of the fractional digits in one of the coordinates. When this happens, the polygon you've created with your mapping tool is not the polygon you've entered into this command and you will not get your desired results. The code implements the pnpoly algorithm. The pnpoly algorithm is copyright © 1970-2003, Wm. Randolph Franklin. Regardless of how many points you specify for your polygon, all 26 points are fed to the algorithm. Due to the mathematical nature of this algorithm, the polygon will not work if vertices of the same polygon lie on opposite sides of the 180 degree longitude divide. Although this command accepts latitude/longitude parameters with up to 15 characters, internally the value is stored as a C float type which has less precision (but requires half the storage size). The float type is capable of storing accuracy commensurate with the GPS receiver's capability, but the queried latitude/longitude values of the AT$PLYFN# command may differ from the input parameters due to this precision limitation.

AT$PLYFN#	Geo Fencing a Polygonal Area
Example	The following commands create a crude diamond-shaped polygon #9 just west of the Enfora offices in Richardson, Texas: AT$PLYFN9=1,32.995498,-96.722064 AT$PLYFN9=2,33.000392,-96.715028 AT$PLYFN9=3,33.008747,-96.722466 AT$PLYFN9=4,33.001330,-96.732982 AT$PLYFN9=5,32.995498,-96.722064 Here's the query response for that polygon definition: AT$PLYFN9? $PLYFN9: PtIdx Latitude Longitude 0 0.0000000 0.0000000 1 32.9954987 -96.7220612 2 33.0003929 -96.7150269 3 33.0087471 -96.7224655 4 33.0013313 -96.7329788 5 32.9954987 -96.7220612 6 0.0000000 0.0000000 OK

AT$PWRSAV Enable Power Save Mode

AT$PWRSAV	Enable Power Save Mode
Command Function	This command allows a user to put the device in low power mode at the <timeout> interval after the Ignition line drops. The ignition line has to be connected per the user manual for this feature to work properly. A user has the capability of getting a notification when the device entering low power mode or returns to normal operating mode.
Syntax Query	AT$PWRSAV=?
Syntax	$PWRSAV: (0-1),(0-500000000),(0-1) OK
Write Syntax	AT$PWRSAV=<ign>,<timeout>,<reg>
Write Response	OK
Read Syntax	AT$PWRSAV?
Read Response	$PWRSAV: 0,0,0 OK
Execute Syntax Execute Response	N/A
Unsolicited Response	N/A
Parameter Values	
<ign>	0 – disable the Ignition feature 1 – enter low power mode after Ignition signal went low and timeout has expired
<timeout>	0 – 65535 seconds. Timeout value after which the unit will enter low power mode. Unit will work in normal mode until the timeout has expired.
<reg>	0 – remain registered with GSM/GPRS network during low power mode 1 – reset modem when entering normal power mode
Notes	An AT$EVENT command has to be set to send a GPS message to the remote host when entering or exiting power save mode. The modem Ignition (switched power) Input must be connected to the vehicles ignition line for this function to work.

Event Tables

Event Engine

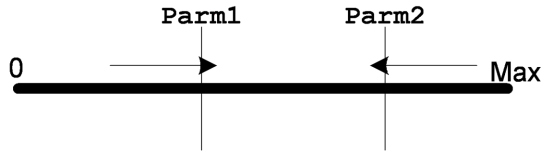
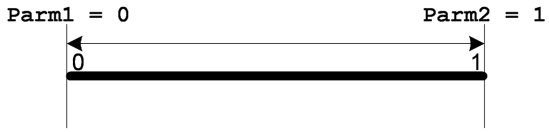
The Enfora Event Engine is a proprietary state machine that can be initiated by one or many triggering input events (which can be logically “ANDed” and/or “ORed”) to generate one or many pre-defined outputs. The Event Engine allows simple, fast development and deployment into end solutions without the need of proprietary development platforms or embedded applications.

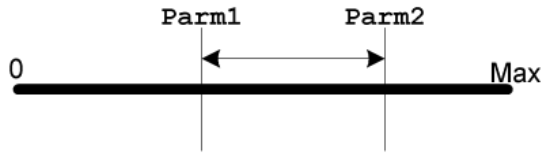
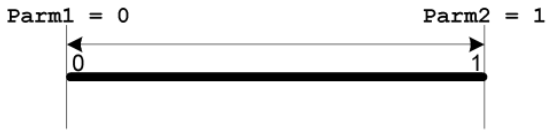
Event Category

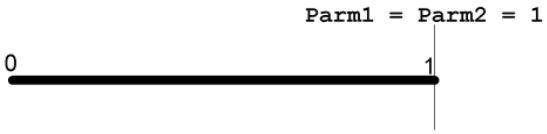
The <event category> parameter defines the actual Input or Output Event number and their valid range for <parm1> and <parm2>.

Event Type

The <event type> parameter defines the type of event: Input or Output. An Input event can be defined as: Transition, Occurrence, or Input. The Output event is executed when input event conditions are met.

Value	Type of event	Description
0	**Transition Trigger**	A transition Trigger is defined as an input condition, defined by <event category>, whose value was previously <parm1> or less is now greater than <parm1> and less than <parm2> or was greater or equal to <parm2> is now less than <parm2> but greater than <parm1>. The output event would be executed when an input <event category> requirements are satisfied or transition to the value set by <parm1> and <parm2> when they are equal. <parm1> should be the min value and <parm2> should be the max value. **Example 1:**  An output event will be executed when the value of an input event exceeds <Parm1> (previously it was <Parm1> or less) or decreases to a value less than <Parm2> (previously it was <Parm2> or greater). **Example 2:**  An output event will be executed when the value of an input event is 0 (previously it was anything else but 0) and <Parm1> along with <Parm2> is set to 0.

Value	Type of event	Description
		Example 3: An output event will be executed when the value of an input event is 1 (previously it was anything else but 1) and <Parm1> along with <Parm2> is set to 1.
1	**Occurrence Trigger**	An Occurrence Trigger is defined as an input condition, defined by <event category>, whose current value is greater than or equal to <parm1> and less than or equal to <parm2>. The output event would be executed when an input <event category> requirements are satisfied or transition to the value set by <parm1> and <parm2> when they are equal. <parm1> should be the min value and <parm2> should be the max value. **Example 1:**  An output event will be executed when the current value of an input event is between <Parm1> and <Parm2> including boundary conditions. **Example 2:**  An output event will be executed when the value of the input event changes from 0 to 1 or vice-versa.

Value	Type of event	Description
		Example 3:  An output event will be executed when the value of the input event is 1 and <Parm1> along with <Parm2> is set to 1.
2	**Input Trigger**	An Input Trigger is defined as an input condition, defined by <event category>, that should be used as a logical AND condition to another input condition defined as Transition Trigger or an Occurrence Trigger. An Output event is not triggered when Input Trigger condition is valid. The input event, defined as Input Trigger, is valid when within the event range defined by <parm1> and <parm2> or when <parm1> and <parm2> are equal.
3	**Output**	An Output event is executed when all input event conditions (defined as Transition Trigger, Occurrence Trigger, or Input Trigger) for that particular <event group> are met. *Output Event Note: ASCII messages sent by the Event Engine have a maximum length of 250 characters (not including the optional header). Any ASCII messages exceding this length are truncated.*

Input Event Table

The following table defines the values for <event category>, <parm1> and <parm2> parameter for input events defined as a Transition Trigger, Occurrence Trigger, or Input Trigger.

Input Event Table			
Event Category	**Parm1**	**Parm2**	**Description**
0	0 or 1	0 or 1	GPI1 – Input #1 0 = Low 1 = High
1	0 or 1	0 or 1	GPO2 – Output #2
2	0 or 1	0 or 1	GPO3 – Output #3; latched
3	0 or 1	0 or 1	GPO4 – LED for GPS user 3 LED (Controlled by MSP, set when AT$GPSCMD is set)
4	0 or 1	0 or 1	GPO5 – Output #5
5	0 or 1	0 or 1	GPO6 – USR1 LED - Registration with default events
6	0 or 1	0 or 1	GPO7 – USR2 LED - GPS lock with default events
7	0 or 1	0 or 1	GPO8 – Input, detects the state of Ignition
8	0 or 1	0 or 1	Modem power up indication
9	0 to 5	0 to 5	Modem GSM registration (see AT+CREG command description for GSM registration status information)
10	0 to 8	0 to 8	Modem GPRS registration (see AT%CGREG command description for GPRS registration status information)
11	0 or 1	0 or 1	Receipt of IP address. 0 = No IP address 1 = Valid IP address obtained

12	1	1	Timer 1 (set by AT$EVTIM1)
13	1	1	Timer 2 (set by AT$EVTIM2)
14	1	1	Timer 3 (set by AT$EVTIM3)
15	1	1	Timer 4 (set by AT$EVTIM4)
16	0 to 1000000	1000000	GPS Distance (unit of measurement is: meters)
17	0 to 250	250	Current Velocity (unit of measurement is: Knots)
18	0-1023	0-1023	ADC 1
19	0-1023	0-1023	ADC 2
20	N/A	N/A	Reserved
21	0 or 1	0 or 1	Geo Fence #1. See AT$GEOFNC command for details on setting a circular geo-fence 0 = Leaving Geofence area 1 = Entering Geofence area
22	0 or 1	0 or 1	Geo Fence #2
23	0 or 1	0 or 1	Geo Fence #3
24	0 or 1	0 or 1	Geo Fence #4
25	0 or 1	0 or 1	Geo Fence #5
26	0 or 1	0 or 1	MT Power Save Event 0 = Exit Power Save Mode 1 = Enter Power Save Mode
27	0 or 1	0 or 1	GPS Status 0 = Invalid GPS data 1 = Valid GPS data
28	1	1	**RTC Alarm Input
29	0 to 1000000	1000000	Invalid GPS data for a period of time (unit of measurement is: seconds)
30	0 to 1000000	1000000	Unit staying Idle in one place (unit of measurement is: seconds)

31	0 or 1	0 or 1	Geo Fence #6. See AT$GEOFNC command for details on setting a circular geo-fence 0 = Leaving Geofence area 1 = Entering Geofence area
32	0 or 1	0 or 1	Geo Fence #7
33	0 or 1	0 or 1	Geo Fence #8
34	0 or 1	0 or 1	Geo Fence #9
35	0 or 1	0 or 1	Geo Fence #10
36	0 or 1	0 or 1	Geo Fence #11
37	0 or 1	0 or 1	Geo Fence #12
38	0 or 1	0 or 1	Geo Fence #13
39	0 or 1	0 or 1	Geo Fence #14
40	0 or 1	0 or 1	Geo Fence #15
41	0 or 1	0 or 1	Geo Fence #16
42	0 or 1	0 or 1	Geo Fence #17
43	0 or 1	0 or 1	Geo Fence #18
44	0 or 1	0 or 1	Geo Fence #19
45	0 or 1	0 or 1	Geo Fence #20
46	0 or 1	0 or 1	Geo Fence #21
47	0 or 1	0 or 1	Geo Fence #22
48	0 or 1	0 or 1	Geo Fence #23
49	0 or 1	0 or 1	Geo Fence #24
50	0 or 1	0 or 1	Geo Fence #25
51	0	0	**Input Event Counter. This event will occur when a counter reaches the maximum number of a selected Input event count.

52	0 or 1	0 or 1	New SMS indication. 0 = SMS message read from SIM 1 = New SMS message received
53	0 to -1	0 to –1	Current Input Event Counter count that can be used as an AND condition with other input events
54	0 or 1	0 or 1	Has the user programmed any geo-fence? Normally this can be found by sending AT$GEOFNC? command and verifying it manually based on the response sent by the device 0 = geo-fence does not exists 1 = at least one geo fence was created
55-59	N/A	N/A	Reserved
60	0 – 9999	0 – 9999	Number of Unsent Messages ($msglogrd count)
61	0 – 100	0 – 100	Memory full percentage ($msglogrd)
62-64	N/A	N/A	Reserved
65	N/A	N/A	Reserved
66	1	1	Timer 5 (set by AT$EVTIM5)
67	1	1	Timer 6 (set by AT$EVTIM6)
68	1	1	Timer 7 (set by AT$EVTIM7)
69	1	1	Timer 8 (set by AT$EVTIM8)
70	0-2147483647	0-2147483647	Current $ODOMETER value
71	N/A	N/A	Reserved
72	0-1	0-1	0 = A GPS overspeed interval has ended 1 = A GPS overspeed interval has begun

73	0 or 1	0 or 1	GPI9 (when selected with AT$RPTADC) This is always GPI9 AND ADC2, even when $RPTADC=1. The $RPTADC setting only controls whether the output msg contains the ADC value or a digital value based on reading ADC2.
74-90	N/A	N/A	Reserved
91	0	2147483647	Trip odometer (distance in meters)
92-99	N/A	N/A	Reserved
100	-2147483648 to 2147483647	-2147483648 to 2147483647	User variable 0
101	-2147483648 to 2147483647	-2147483648 to 2147483647	User variable 1
102	-2147483648 to 2147483647	-2147483648 to 2147483647	User variable 2
103	-2147483648 to 2147483647	-2147483648 to 2147483647	User variable 3
104	-2147483648 to 2147483647	-2147483648 to 2147483647	User variable 4
105	-2147483648 to 2147483647	-2147483648 to 2147483647	User variable 5
106	-2147483648 to 2147483647	-2147483648 to 2147483647	User variable 6
107	-2147483648 to 2147483647	-2147483648 to 2147483647	User variable 7
108	-2147483648 to 2147483647	-2147483648 to 2147483647	User variable 8
109	-2147483648 to 2147483647	-2147483648 to 2147483647	User variable 9
110	N/A	N/A	Reserved

111	0 or 1	0 or 1	GSM_JAMMING_STATE Current state of the GSM jamming detection algorithm. JAMMED = 1 CLEAR = 0
112-118	N/A	N/A	Reserved
119	0 or 1	0 or 1	0 = An alternate GPS overspeed interval has ended. 1 = An alternate GPS overspeed interval has begun.
120	0 or 1	0 or 1	Polygon Geofence #0. See $ATPLYFN# for details on setting a polygonal geofence. 0 = Leaving Geofence area 1 = Entering Geofence area
121	0 or 1	0 or 1	Polygon Geofence #1
122	0 or 1	0 or 1	Polygon Geofence #2
123	0 or 1	0 or 1	Polygon Geofence #3
124	0 or 1	0 or 1	Polygon Geofence #4
125	0 or 1	0 or 1	Polygon Geofence #5
126	0 or 1	0 or 1	Polygon Geofence #6
127	0 or 1	0 or 1	Polygon Geofence #7
128	0 or 1	0 or 1	Polygon Geofence #8
129	0 or 1	0 or 1	Polygon Geofence #9

130	0 - 2147483647	0 - 2147483647	Bit-mapped value indicating if/why GPS data is being invalidated via $GPSQUAL filtering: 0 = data is good 1 = delay interval in progress due to earlier detection of bad uncertainty, heading change, speed, distance and/or acceleration 2 = velocity fix flag invalid 4 = uncertainty value > threshold 8 = $GPGSA fix type not 3D 16 = $GPGSA HDOP < threshold 32 = GPS date invalid 64 = GPS time interval suspect 128 = speed > threshold 256 = acceleration over one sec > threshold 512 = heading change over one sec > threshold 1024 = # of satellites being tracked < threshold 2048 = distance over one sec > threshold
131	N/A	N/A	Reserved
132	0 or 1	0 or 1	Polygon Geofence #10
133	0 or 1	0 or 1	Polygon Geofence #11
134	0 or 1	0 or 1	Polygon Geofence #12
135	0 or 1	0 or 1	Polygon Geofence #13
136	0 or 1	0 or 1	Polygon Geofence #14
137	0 or 1	0 or 1	Polygon Geofence #15
138	0 or 1	0 or 1	Polygon Geofence #16
139	0 or 1	0 or 1	Polygon Geofence #17
140	0 or 1	0 or 1	Polygon Geofence #18

141	0 or 1	0 or 1	Polygon Geofence #19
142	0 or 1	0 or 1	Polygon Geofence #20
143	0 or 1	0 or 1	Polygon Geofence #21
144	0 or 1	0 or 1	Polygon Geofence #22
145	0 or 1	0 or 1	Polygon Geofence #23
146	0 or 1	0 or 1	Polygon Geofence #24
147	1	1	This event is triggered at the completion of a FOTA upgrade after the modem has rebooted with the new firmware (old firmware if the upgrade failed).
148	0 or 1	0 or 1	Accelerometer Filter X1 limit exceeded 0 = acceleration limit not exceeded 1 = acceleration limit exceeded
149	0 or 1	0 or 1	Accelerometer Filter X2 limit exceeded 0 = acceleration limit not exceeded 1 = acceleration limit exceeded
150	0 or 1	0 or 1	Accelerometer Any Motion event 0 = 'Any motion' did not occur 1 = 'any motion' occurred
151	0 or 1	0 or 1	Accelerometer Filter Y1 limit exceeded 0 = acceleration limit not exceeded 1 = acceleration limit exceeded
152	0 or 1	0 or 1	Accelerometer Filter Y2 limit exceeded 0 = acceleration limit not exceeded 1 = acceleration limit exceeded
153	0 or 1	0 or 1	Accelerometer Filter Z1 limit exceeded 0 = acceleration limit not exceeded 1 = acceleration limit exceeded

154	0 or 1	0 or 1	Accelerometer Filter Z2 limit exceeded 0 = acceleration limit not exceeded 1 = acceleration limit exceeded
155-159	N/A	N/A	Reserved
160	1	1	Accelerometer MGC status
161-182	N/A	N/A	Reserved
183	1	1	Timer 9 (set by AT$EVTIM9)
184	1	1	Timer 10 (set by AT$EVTIM10)
185-197	NA	NA	Reserved
198	0 or 1	0 or 1	Excessive Acceleration
199	0 or 1	0 or 1	Excessive Deceleration
200	0 or 1	0 to 43200	Triggered when RTC reaches the target check in time (set by AT$CHKIN)

Output Event Table

The below table defines the values for <event category>, <parm1> and <parm2> parameter for output events defined as Output.

Output Event Table			
Event Category	Parm1	Parm2	Description
0	0	0	Reserved
9	0	0	Set GPO #2 to Low (0)
10	0	0	Set GPO #3 to Low (0)
11	0	0	Set GPIO #4 configured as Output to Low (0)
12	0	0	Reserved
13	0	0	Set GPIO #6 configured as Output to Low (0)
14	0	0	Set GPIO #7 configured as Output to Low (0)
15	0	0	Reserved
16	0	0	Reserved
17	0	0	Set GPO #2 to High (1)
18	0	0	Set GPO #3 to High (1)
19	0	0	Set GPIO #4 configured as Output to High (1)
20	0	0	Reserved
21	0	0	Set GPIO #6 configured as Output to High (1)
22	0	0	Set GPIO #7 configured as Output to High (1)
23	0	0	Reserved
24	0	0	Reserved
25	0	0	Toggle GPO #2
26	0	0	Toggle GPO #3
27	0	0	Toggle GPIO #4 configured as Output
28	0	0	Reserved

29	0	0	Toggle GPIO #6 configured as Output
30	0	0	Toggle GPIO #7 configured as Output
31	0	0	Reserved
32	See GPIO Flash Table		Reserved
33			Flash GPO #2
34			Flash GPO #3
35			Flash GPIO #4 configured as Output
36			Reserved
37			Flash GPIO #6 configured as Output
38			Flash GPIO #7 configured as Output
39			Reserved

40	0 to 214783647	See Bit-Field Table	Generate and transmit one UDP Message to first IP address listed in $FRIEND command and port number listed in $UDPAPI command based on Parm1 and Parm2 values
41			Generate and transmit a UDP message with Acknowledge. This message is controlled by $ACKTM command for number of retries sent. This message has to be acknowledged to avoid sending of retries.
42			Generate and transmit one UDP Message to all IP address listed in $FRIEND command and port number listed in $UDPAPI command based on Parm1 and Parm2 values
43	1 – 8	0	Resets the timer (Timer #1 - Timer #8) specified by Parm1 to the time (in seconds) specified by Parm2. Parm2, when set to 0, resets the timer to the time last set by $EVTIMx command or previous output event 43 execution. A value other than 0 would set the timer to expire at the new specified interval (e.g. xx,3,43,1,180 would set timer 1 to expire in 180 seconds). When used with a value other than 0, this is equivalent to invoking $EVTIMx directly from the event engine and subsequent AT&F commands will save the new value to flash.

44	1 – 25	0	Execute AT command stored at index number of the $STOATEV command. Parm1 identifies the index number.
45	0 to 2147483647	See Bit-Field Table	Sends data over SMS to All SMS destination addresses configured via $SMSDA command. (For select $SMSDA entries, see event categories 54-58)
46	N/A	N/A	Reserved
47	0	0 to –1	Input Event Counter
48	0	0 to –1	Input Event Counter reset to value stated by parm2
49	1 – 25	0 - 1000000	Set geo-fence specified by parm1 to current latitude & longitude with radius specified by parm2
50	0 – 57	0 to –1	Emulate AT$EVTEST command via event engine. Parm1 is the input event number while Parm2 is the value to emulate for the input event
51	N/A	N/A	Reserved
52	0 to –1	See Bit-Field Table	Generate and transmit one TCP/IP Message to IP address & port number listed by $FRIEND command based on Parm1 and Parm2 values

<table>
<tr>
<td>53</td>
<td>0 – 99</td>
<td>0 – 8</td>
<td>Sets periodic RTC alarm in minutes, hours, days, or months.

Parm1 indicates the frequency with which to generate the message. Parm2 indicates the time-unit used.

Parm2 values:

1 = minutes

2 = hours

4 = days

8 = months

For example:
<table>
<tr><th>Parm1</th><th>Parm2</th><th>Result - RTC Alarm occurs every [parm1] [parm2]</th></tr>
<tr><td>1</td><td>1</td><td>RTC Alarm occurs every 1 minute</td></tr>
<tr><td>3</td><td>2</td><td>RTC Alarm occurs every 3 hours</td></tr>
<tr><td>1</td><td>4</td><td>RTC Alarm occurs every 1 day</td></tr>
<tr><td>6</td><td>8</td><td>RTC Alarm occurs every 6 months</td></tr>
</table>
</td>
</tr>
</table>

54	0 to 2147483647	See Bit-Field Table	Sends data over SMS to the first indexed SMS destination address configured via $SMSDA command.
55	0 to 2147483647		Sends data over SMS to the second indexed SMS destination address configured via $SMSDA command.
56	0 to 2147483647		Sends data over SMS to the third indexed SMS destination address configured via $SMSDA command.
57	0 to 2147483647		Sends data over SMS to the fourth indexed SMS destination address configured via $SMSDA command.
58	0 to 2147483647		Sends data over SMS to the fifth indexed SMS destination address configured via $SMSDA command.
59	N/A	N/A	Reserved
60	0 to -1	See Bit-Field Table	Generate and transmit message to main serial port based on Parm1 and Parm2 values in ASCII format only.
61-124	N/A	N/A	Reserved

125	0 – 9	-2147483648 to 2147483647	Sets user variable indicated by parm1 to value of parm2 (for example, if parm1 is 7 and parm2 is 50, this output event would set user variable 7 to 50)
126	0 – 9	-2147483648 to 2147483647	Increments user variable indicated by parm1 by value of parm2 (for example, if parm1 is 4, parm2 is 100, and user variable 4 was 200 prior to this event, user variable 4 would be incremented to 300 by this event)
127	0 – 9	-2147483648 to 2147483647	Decrements user variable indicated by parm1 by value of parm2 (for example, if parm1 is 9, parm2 is 10, and user variable 9 was 50 prior to this event, user variable 9 would be decremented to 40 by this event)
128	0 – 9	-2147483648 to 2147483647	Copies value of a system variable into user variable indicated by parm1. Parm2 is used as an index to determine the system variable that will be copied (see User Variable Index Table)
129-137	N/A	N/A	Reserved
138	1 – 20	N/A	Play audio file that is stored in the FFS. The audio files must be named 1-20.
139-146	N/A	N/A	Reserved
147	0	1	Save circular GPS AI2 buffer to /cst/ai2log in FFS (used to debug unexpected GPS behavior).
148-151	N/A	N/A	Reserved
152	0	0	Send UDP message to the check-in servers specified in the $FRIEND list. This event sends a wakeup style message to every server in the friends list with usage=4.

User Variable Index Table

This table is used only with Output Event 128

Parm2	System Variable Copied to User Variable
	(For example, AT$EVENT=99,3,128,3,9 would copy value of Input Event 9 (GSM registration status) into User Variable 3). NOTE: All the following system variables are not supported by all devices. Ensure your device supports the system variable before attempting to use it with user variables. For Serving Cell and Neighbor Cell values, see GSM0000GN012 – Engineering Mode Manual for details of the %EM command.
-856	#define N4A_USRVAR_ACCEL_NXFM_Z (-865) ;/ Accelerometer non-transformed Z axis value
-864	#define N4A_USRVAR_ACCEL_NXFM_Y (-864) ;/ Accelerometer non-transformed Y axis value
-863	#define N4A_USRVAR_ACCEL_NXFM_X (-863) ;/ Accelerometer non-transformed X axis value
-862	#define N4A_USRVAR_ACCEL_MGC_COUNT (-862) ;/ MagiCal count
-861	#define N4A_USRVAR_ACCEL_MGC_STATE (-861) ;/ MagiCal state
-860	#define N4A_USRVAR_ACCEL_MGC_MODE (-860) ;/ MagiCal mode setting
-474 to -859	Reserved
-473	Copies Neighbor Cell 5 signal strength. Equivalent to AT%EM=2,3
-472	Copies Neighbor Cell 5 absolute radio frequency channel number (ARFCN). Equivalent to AT%EM=2,3
-471	Copies Neighbor Cell 5 cell ID. Equivalent to AT%EM=2,3
-470	Copies Neighbor Cell 5 location area code. Equivalent to AT%EM=2,3
-469 to -464	Reserved
-463	Copies Neighbor Cell 4 signal strength. Equivalent to AT%EM=2,3
-462	Copies Neighbor Cell 4 absolute radio frequency channel number (ARFCN). Equivalent to AT%EM=2,3

-461	Copies Neighbor Cell 4 cell ID. Equivalent to AT%EM=2,3
-460	Copies Neighbor Cell 4 location area code. Equivalent to AT%EM=2,3
-459 to -454	Reserved
-453	Copies Neighbor Cell 3 signal strength. Equivalent to AT%EM=2,3
-452	Copies Neighbor Cell 3 absolute radio frequency channel number (ARFCN). Equivalent to AT%EM=2,3
-451	Copies Neighbor Cell 3 cell ID. Equivalent to AT%EM=2,3
-450	Copies Neighbor Cell 3 location area code. Equivalent to AT%EM=2,3
-449 to -444	Reserved
-443	Copies Neighbor Cell 2 signal strength. Equivalent to AT%EM=2,3
-442	Copies Neighbor Cell 2 absolute radio frequency channel number (ARFCN). Equivalent to AT%EM=2,3
-441	Copies Neighbor Cell 2 cell ID. Equivalent to AT%EM=2,3
-440	Copies Neighbor Cell 2 location area code. Equivalent to AT%EM=2,3
-439 to -434	Reserved
-433	Copies Neighbor Cell 1 signal strength. Equivalent to AT%EM=2,3
-432	Copies Neighbor Cell 1 absolute radio frequency channel number (ARFCN). Equivalent to AT%EM=2,3
-431	Copies Neighbor Cell 1 cell ID. Equivalent to AT%EM=2,3
-430	Copies Neighbor Cell 1 location area code. Equivalent to AT%EM=2,3
-429 to -424	Reserved
-423	Copies Neighbor Cell 0 signal strength. Equivalent to AT%EM=2,3
-422	Copies Neighbor Cell 0 absolute radio frequency channel number (ARFCN). Equivalent to AT%EM=2,3
-421	Copies Neighbor Cell 0 cell ID. Equivalent to AT%EM=2,3
-420	Copies Neighbor Cell 0 location area code. Equivalent to AT%EM=2,3
-419 to -407	Reserved
-406	Copies Serving Cell timing advance. Equivalent to AT%EM=2,1
-405	Copies Serving Cell signal strength. Equivalent to AT%EM=2,1

-404	Copies Serving Cell absolute radio frequency channel number (ARFCN). Equivalent to AT%EM=2,1
-403	Copies Serving Cell cell ID. Equivalent to AT%EM=2,1
-402	Copies Serving Cell location area code. Equivalent to AT%EM=2,1
-401	Copies Serving Cell MNC (0x00MMNNCC, where MM, NN, and CC are the hex values of the ASCII representations of the MNC). Equivalent to AT%EM=2,4
-400	Copies Serving Cell MCC (0x00MMCCcc, where MM, CC, and cc are the hex values of the ASCII representations of the MCC) . Equivalent to AT%EM=2,4
-399 to -303	Reserved
-302	Copies Software version (for example, if version is 1.1.1.8, value would be 0x00001118)
-301	Copies product ID
-300	Copies $usrval value
-299 to -42	Reserved
-41	Copies GPIO value status where LSB represents GPIO0. For example: 0x00165432 2=binary 0010 (so GPIO1=1; GPIO0, GPIO2, GPIO3=0) 3=binary 0011 (so GPIO4,GPIO5=1; GPIO6,GPIO7=0) 4=binary 0100 (so GPIO10=1; GPIO8,GPIO9,GPIO11=0) 5=binary 0101 (so GPIO12,GPIO14=1; GPIO13,GPIO15=0) 6=binary 0110 (so GPIO17,GPIO18=1; GPIO16,GPIO19=0) 1=binary 0001 (so GPIO20=1) Note: GPIO numbers >8 are not supported on the MT-Gµ

-40	Copies GPIO direction status where LSB represents GPIO0 (1=input,0-output). For example: 0x00165432 2=binary 0010 (so GPIO1=input; GPIO0, GPIO2, GPIO3= outputs) 3=binary 0011 (so GPIO4,GPIO5=inputs; GPIO6,GPIO7=outputs) 4=binary 0100 (so GPIO10=input; GPIO8,GPIO9,GPIO11=outputs) 5=binary 0101 (so GPIO12,GPIO14=inputs; GPIO13,GPIO15=outputs) 6=binary 0110 (so GPIO17,GPIO18=inputs; GPIO16,GPIO19=outputs) 1=binary 0001 (so GPIO20=input) Note: GPIO numbers >8 are not supported on the MT-Gµ
-39 to -22	Reserved
-21	Copies current RTC time: 0x00HHMMSS where HH = hour (0-23), MM = minute (0-59), SS = second (0-59)
-20	Copies current RTC date: 0x00YYMMDD where YY = last two digits of year (00-99), MM = month (1-12), DD = day of month (1-31)
-19 to -9	Reserved
-8	Copies current count of event timer 8 in seconds (equivalent to $EVTIMQRY=8)
-7	Copies current count of event timer 7 in seconds (equivalent to $EVTIMQRY=7)
-6	Copies current count of event timer 6 in seconds (equivalent to $EVTIMQRY=6)
-5	Copies current count of event timer 5 in seconds (equivalent to $EVTIMQRY=5)
-4	Copies current count of event timer 4 in seconds (equivalent to $EVTIMQRY=4)
-3	Copies current count of event timer 3 in seconds (equivalent to $EVTIMQRY=3)
-2	Copies current count of event timer 2 in seconds (equivalent to $EVTIMQRY=2)
-1	Copies current count of event timer 1 in seconds (equivalent to $EVTIMQRY=1)
0	Copies value of Input Event 0 (GPIO1). 0 = Low 1 = High

1	Copies value of Input Event 1 (GPIO2). 0 = Low 1 = High
2	Copies value of Input Event 2 (GPIO3). 0 = Low 1 = High
3	Copies value of Input Event 3 (GPIO4). 0 = Low 1 = High
4	Copies value of Input Event 4 (GPIO5). 0 = Low 1 = High
5	Copies value of Input Event 5 (GPIO6). 0 = Low 1 = High
6	Copies value of Input Event 6 (GPIO7). 0 = Low 1 = High
7	Copies value of Input Event 7 (GPIO8). 0 = Low 1 = High
8	Copies value of Input Event 8 (modem power up indication). Always 1.
9	Copies value of Input Event 9 (modem GSM registration). See AT+CREG command description for GSM registration status information.
10	Copies value of Input Event 10 (modem GPRS registration). See AT%CGREG command for GPRS registration status information.

11	Copies value of Input Event 11 (Receipt of IP address). 0 = No IP address 1 = Valid IP address obtained
12	Copies value of Input Event 12 (Timer 1 status). 0 = Timer not expired 1 = Timer expired
13	Copies value of Input Event 13 (Timer 2 status). 0 = Timer not expired 1 = Timer expired
14	Copies value of Input Event 14 (Timer 3 status). 0 = Timer not expired 1 = Timer expired
15	Copies value of Input Event 15 (Timer 4 status). 0 = Timer not expired 1 = Timer expired
16	Copies value of Input Event 16 (GPS distance in meters)
17	Copies value of Input Event 17 (Maximum velocity in knots)
18	Copies value of Input Event 18 (ADC1 status)
19 to 20	Reserved
21	Copies value of Input Event 21 (Geofence #1) 0 = Leaving geofence area 1 = Entering geofence area
22	Copies value of Input Event 22 (Geofence #2) 0 = Leaving geofence area 1 = Entering geofence area
23	Copies value of Input Event 23 (Geofence #3) 0 = Leaving geofence area 1 = Entering geofence area

24	Copies value of Input Event 24 (Geofence #4) 0 = Leaving geofence area 1 = Entering geofence area
25	Copies value of Input Event 25 (Geofence #5) 0 = Leaving geofence area 1 = Entering geofence area
26	Copies value of Input Event 26 (MT Power Save Event) 0 = Exit Power Save Mode 1 = Enter Power Save Mode
27	Copies value of Input Event 27 (GPS status) 0 = Invalid GPS data 1 = Valid GPS data
28	Copies value of Input Event 28 (RTC Alarm Input)
29	Copies value of Input Event 29 (Invalid GPS data for a period of seconds)
30	Copies value of Input Event 30 (Unit staying Idle in one place for a period of seconds)
31	Copies value of Input Event 31 (Geofence #6) 0 = Leaving geofence area 1 = Entering geofence area
32	Copies value of Input Event 32 (Geofence #7) 0 = Leaving geofence area 1 = Entering geofence area
33	Copies value of Input Event 33 (Geofence #8) 0 = Leaving geofence area 1 = Entering geofence area
34	Copies value of Input Event 34 (Geofence #9) 0 = Leaving geofence area 1 = Entering geofence area

35	Copies value of Input Event 35 (Geofence #10) 0 = Leaving geofence area 1 = Entering geofence area
36	Copies value of Input Event 36 (Geofence #11) 0 = Leaving geofence area 1 = Entering geofence area
37	Copies value of Input Event 37 (Geofence #12) 0 = Leaving geofence area 1 = Entering geofence area
38	Copies value of Input Event 38 (Geofence #13) 0 = Leaving geofence area 1 = Entering geofence area
39	Copies value of Input Event 39 (Geofence #14) 0 = Leaving geofence area 1 = Entering geofence area
40	Copies value of Input Event 40 (Geofence #15) 0 = Leaving geofence area 1 = Entering geofence area
41	Copies value of Input Event 41 (Geofence #16) 0 = Leaving geofence area 1 = Entering geofence area
42	Copies value of Input Event 42 (Geofence #17) 0 = Leaving geofence area 1 = Entering geofence area
43	Copies value of Input Event 43 (Geofence #18) 0 = Leaving geofence area 1 = Entering geofence area

44	Copies value of Input Event 44 (Geofence #19) 0 = Leaving geofence area 1 = Entering geofence area
45	Copies value of Input Event 45 (Geofence #20) 0 = Leaving geofence area 1 = Entering geofence area
46	Copies value of Input Event 46 (Geofence #21) 0 = Leaving geofence area 1 = Entering geofence area
47	Copies value of Input Event 47 (Geofence #22) 0 = Leaving geofence area 1 = Entering geofence area
48	Copies value of Input Event 48 (Geofence #23) 0 = Leaving geofence area 1 = Entering geofence area
49	Copies value of Input Event 49 (Geofence #24) 0 = Leaving geofence area 1 = Entering geofence area
50	Copies value of Input Event 50 (Geofence #25) 0 = Leaving geofence area 1 = Entering geofence area
51	Copies value of Input Event 51 (Input Event Counter)
52	Copies value of Input Event 52 (New SMS indication) 0 = SMS message read from SIM 1 = New SMS message received
53	Copies value of Input Event 53 (Current Input Event Counter count that can be used as an AND condition with other input events)

54	Copies value of Input Event 54 (Does any geofence exist?) 0 = geofence does not exist 1 = at least one geofence was created
55 to 64	Reserved
65	Copies value of Input Event 64 (Receipt of incoming call with Call Identifier matching one of the numbers configured via the $EVCID command)
66	Copies value of Input Event 66 (Timer 5 status). 0 = Timer not expired 1 = Timer expired
67	Copies value of Input Event 67 (Timer 6 status). 0 = Timer not expired 1 = Timer expired
68	Copies value of Input Event 68 (Timer 7 status). 0 = Timer not expired 1 = Timer expired
69	Copies value of Input Event 69 (Timer 8 status). 0 = Timer not expired 1 = Timer expired
70	Copies value of Input Event 70 (Odometer in meters)
71	Reserved
72	Copies value of Input Event 72 (GPS overspeed) 0 = interval has ended 1 = interval has begin
73 to 90	Reserved
91	Copies value of Input Event 91 (Trip odometer in meters)
92 to 99	Reserved
100	Copies value of Input Event 100 (User variable 0)
101	Copies value of Input Event 101 (User variable 1)

102	Copies value of Input Event 102 (User variable 2)
103	Copies value of Input Event 103 (User variable 3)
104	Copies value of Input Event 104 (User variable 4)
105	Copies value of Input Event 105 (User variable 5)
106	Copies value of Input Event 106 (User variable 6)
107	Copies value of Input Event 107 (User variable 7)
108	Copies value of Input Event 108 (User variable 8)
109	Copies value of Input Event 109 (User variable 9)

GPIO Flash Table

Parm1	Parm2
Bits 16 – 31 determine the low signal state while bits 0 – 15 determine the high signal state. A value of 0 for bits 16 – 31 indicates the GPIO will remain in low signal state for the same amount of time as the high signal state (50% duty cycle). The high or low states are measured in multiples of ¼ seconds. The toggle count is set by Parm2	The flashing GPIO event will cause the GPIO output state to toggle at time 0 to the opposite state prior to starting the GPIO output flash event processing. This counts as toggle #1. An even number of toggle count will force a final state which is the same as the initial state. An odd number of toggle count will force the final state to be opposite of the initial GPIO output condition. 0 = toggle forever.

Note:

When flashing GPIO 3 it must be longer than one second.

Bit-Field Tables

Bit-Field Table Selection

Use the table below to determine which of the four bit-field tables (0-3) to use for the Parm2 value.

Bit-Field Table Selection		
Bit 31	**Bit 30**	**Description**
0	0	Table selector 0. Format message based on Parm2 values using Message Format Table 0 (legacy format)
0	1	Table selector 1. Format message based on Parm2 values using Message Format Table 1.
1	0	Table selector 2. Format message based on Parm2 values using Message Format Table 2.
1	1	Table selector 3. Format message based on Parm2 values using Message format Table 3.

Bit-Field Table 0 – Legacy (0,0)

The Parm2 value is obtained as a result of selecting individual bit-fields from the table below.

Bit-Field Table 0 – Legacy (0,0)	
Parm2	**Description**
Bit 0:	1 = send all data generated as a result of this table in Binary format 0 = send all data generated as a result of this table in ASCII format
Bit 1:	1 = add parm 1 data to UDP message (4 – bytes in Binary format, 11 – bytes of data in ASCII format) 0 = do not add parm1 data to outbound UDP message
Bit 2:	1 = add $MDMID value (22 – bytes of ASCII data – irrespective of Bit– 0 setting) 0 = do not add $MDMID value
Bit 3:	1 = add $IOCFG and $IOGPA (GPIO direction and data) in ASCII-HEX format (2 – bytes in Binary format, 6 – bytes in ASCII format) 0 = do not add GPIO direction and data value.

Bit 4:	1 = add ADC1 value (2 bytes binary or 5 bytes ASCII) 0 = do not add ADC1 value
Bit 5:	1 = add ADC2 value or GPI9 status data(2 bytes binary or 5 bytes ASCII) 0 = do not add ADC2 value or GPI9 status data
Bit 6:	1 =Message is stored in non-volatile memory until it can be sent, regardless of network status. 0 = Code checks network status before storing message in non-volatile memory. If it appears that the message can be sent out immediately (network status is clear and message queue has few or no messages pending), the message is stored in the non-volatile message queue until it can be sent. Otherwise, the message is deleted.
Bit 7:	1 = add input <event category> number (1 – byte in binary format, 3 – bytes in ASCII format) 0 = do not add input <event category> number
Bit 8:	1 = add GPS data (3 – bytes of Date information in Binary format or up to 80 – bytes of $GPGGA NMEA message if Bit-0 is set to 0) 0 = do not add this particular field of GPS data
Bit 9:	1 = add 1-byte of STATUS information in Binary 0 = do not add this particular field of GPS data
Bit 10:	1 = add GPS data (3 – bytes of Latitude information in Binary format or up to 80 – bytes of $GPGSA NMEA message if Bit-0 is set to 0) 0 = do not add this particular field of GPS data
Bit 11:	1 = add GPS data (4 – bytes of Longitude information in Binary format or up to two 80 – bytes of $GPGSV NMEA message if Bit-0 is set to 0) 0 = do not add this particular field of GPS data
Bit 12:	1 = add GPS data (2 – bytes of Velocity information in Binary format or up to 80 – bytes of $GPRMC NMEA message if Bit-0 is set to 0) 0 = do not add this particular field of GPS data
Bit 13:	1 = add 2-bytes of HEADING information in Binary 0 = do not add this particular field of GPS data
Bit 14:	1 = add GPS data (3 – bytes of Time information in Binary format or 0 bytes if Bit-0 is set to 0) 0 = do not add this particular field of GPS data
Bit 15:	1 = add GPS data (3 – bytes of Altitude information in Binary format or 0 bytes if Bit-0 is set to 0) 0 = do not add this particular field of GPS data

Bit 16:	1 = add GPS data (1 – byte of Number Of Satellites In View information in Binary format or 0 bytes if Bit-0 is set to 0) 0 = do not add this particular field of GPS data
Bit 17:	Reserved
Bit 18:	1 = send this OTA message via SMS when GPRS services is not available 0 = send this OTA message via GPRS only
Bit 19:	1 = send Last Valid GPS data if current data is invalid 0 = send current GPS data – valid or invalid
Bit 20:	1 = add Odometer reading (4 – bytes of Odometer information in Binary format or 11 – bytes if Bit-0 is set to 0) 0 = do not add this particular field of GPS data
Bit 21:	1 = add RTC time (6 – bytes of RTC time in Binary format or 13 – bytes if Bit-0 is set to 0) 0 = do not add RTC time with GPS data
Bit 22:	1 = Replace/append modem ID field with 10-byte modem ID (including one leading and one ending space character) if bit-0 is set to 0. Replace/append it with 8-bytes long modem ID value if bit-0 is set to 1 (no leading or ending space characters in binary mode.) (NOTE: bit-22 setting overrides bit-2 setting) 0 = Sent the modem ID as defined by Bit-2
Bit 23:	Reserved
Bit 24:	1 = add GPS overspeed data (6 – bytes of Odometer information in Binary format or 6 to 18 – bytes if Bit-0 is set to 0). Binary format: xxyyzz: xx is speed specified by AT$GPSOSI (unit: knots); yy is the maximum speed incurred during the interval (unit: knots, 1/10 knot accuracy); zz is the interval duration (unit: seconds); ASCII format: " x y z": space delineated, length of each field varies with its value 0 = do not add this particular field of GPS data

Bit 25:	1 = Add cell information as follows (see GSM0000TN012 - Engineering Mode Manual for details of the %EM command): If Binary format (Bit0=1) is selected, please refer to the “Bit 25 Binary Format” table If ASCII format (Bit0=0) is selected please refer to the “Bit 25 ASCII Format” table 0 = Do not add cell information
Bits 26 – 29:	Reserved
Bits 30 – 31:	00 = This mask identifier signifies output event mask 0

Bit 25 Binary Format Table

Bit 25 Binary Format Table		
	Position	**Description**
MCC	0-2	3 bytes - equivalent to mnc digits returned from AT%EM=2,4
MNC	3-4	2 bytes - equivalent to lac returned from AT%EM=2,1
Serving Cell LAC	5-6	2 bytes - equivalent to cell_id returned from AT%EM=2,1
Serving Cell CellID	7-8	2 bytes - equivalent to arfcn returned from AT%EM=2,1
Serving Cell ARFCN	9	1 byte - equivalent to rxlev returned from AT%EM=2,1 in Idle mode or rxlev_f in Dedicated mode
Serving Cell signal strength	10	1 byte - equivalent to tav returned in AT%EM=2,1
Serving Cell timing advance	11-12	2 bytes - equivalent to lac_nc[0] returned from AT%EM=2,3
Neighbor Cell 0 LAC	13-14	2 bytes - equivalent to cell_id_nc[0] returned from AT%EM=2,3
Neighbor Cell 0 CellID	15-16	2 bytes - equivalent to afrcn_nc[0] returned from AT%EM=2,3
Neighbor Cell 0 ARFCN	17	1 byte - equivalent to rxlev_nc[0] returned from AT%EM2,3
Neighbor Cell 0 rxlev	18	1 byte - equivalent to rxlev_nc[0] returned from AT%EM2,3
Neighbor Cell 1 LAC	19-20	2 bytes - equivalent to lac_nc[1] returned from AT%EM=2,3
Neighbor Cell 1 CellID	21-22	2 bytes - equivalent to cell_id_nc[1] returned from AT%EM=2,3
Neighbor Cell 1 ARFCN	23-24	2 bytes - equivalent to afrcn_nc[1] returned from AT%EM=2,3

Neighbor Cell 1 rxlev	25	1 byte - equivalent to rxlev_nc[1] returned from AT%EM2,3
Neighbor Cell 2 LAC	26-27	2 bytes - equivalent to lac_nc[2] returned from AT%EM=2,3
Neighbor Cell 2 CellID	28-29	2 bytes - equivalent to cell_id_nc[2] returned from AT%EM=2,3
Neighbor Cell 2 ARFCN	30-31	2 bytes - equivalent to afrcn_nc[2] returned from AT%EM=2,3
Neighbor Cell 2 rxlev	32	1 byte - equivalent to rxlev_nc[2] returned from AT%EM2,3
Neighbor Cell 3 LAC	33-34	2 bytes - equivalent to lac_nc[3] returned from AT%EM=2,3
Neighbor Cell 3 CellID	35-36	2 bytes - equivalent to cell_id_nc[3] returned from AT%EM=2,3
Neighbor Cell 3 ARFCN	37-38	2 bytes - equivalent to afrcn_nc[3] returned from AT%EM=2,3
Neighbor Cell 3 rxlev	39	1 byte - equivalent to rxlev_nc[3] returned from AT%EM2,3
Neighbor Cell 4 LAC	40-41	2 bytes - equivalent to lac_nc[4] returned from AT%EM=2,3
Neighbor Cell 4 CellID	42-43	2 bytes - equivalent to cell_id_nc[4] returned from AT%EM=2,3
Neighbor Cell 4 ARFCN	44-45	2 bytes - equivalent to afrcn_nc[4] returned from AT%EM=2,3
Neighbor Cell 4 rxlev	46	1 byte - equivalent to rxlev_nc[4] returned from AT%EM2,3
Neighbor Cell 5 LAC	47-48	2 bytes - equivalent to lac_nc[5] returned from AT%EM=2,3
Neighbor Cell 5 CellID	49-50	2 bytes - equivalent to cell_id_nc[5] returned from AT%EM=2,3
Neighbor Cell 5 ARFCN	51-52	2 bytes - equivalent to afrcn_nc[5] returned from AT%EM=2,3
Neighbor Cell 5 rxlev	53	1 byte - equivalent to rxlev_nc[5] returned from AT%EM2,3

Bit 25 ASCII Format Table

Bit 25 ASCII Format Table

If ASCII format (Bit0=0) is selected: Variable length string is appended to message with semicolons separating cells Serving Cells and Neighbor Cells) and commas separating the fields within a cell as follows:

mcc,mnc,sc_lac,sc_cell_id,sc_arfcn,sc_rxlev,sc_tav;

lac_nc0,cell_id_nc0,arfcn_nc0,rxlev_nc0;

lac_nc1,cell_id_nc1,arfcn_nc1,rxlev_nc1;

lac_nc2,cell_id_nc2,arfcn_nc2,rxlev_nc2;

lac_nc3,cell_id_nc3,arfcn_nc3,rxlev_nc3;

lac_nc4,cell_id_nc4,arfcn_nc4,rxlev_nc4;

lac_nc5,cell_id_nc5,arfcn_nc5,rxlev_nc5

	Postition	**Description**
mcc	1	equivalent to lac returned from AT%EM=2,1
mnc	2	equivalent to cell_id returned from AT%EM=2,1
sc_lac	3	equivalent to arfcn returned from AT%EM=2,1
sc_cell_id	4	equivalent to rxlev returned from AT%EM=2,1 in Idle mode or rxlev_f in Dedicated mode
sc_arfcn	5	equivalent to tav returned from AT%EM=2,1
sc_rxlev	6	equivalent to lac_nc[0] returned from AT%EM=2,3
sc_tav	7	equivalent to cell_id_nc[0] returned from AT%EM=2,3
lac_nc0	8	equivalent to arfcn_nc[0] returned from AT%EM=2,3
cell_id_nc0	9	equivalent to rxlev_nc[0] returned from AT%EM=2,3
arfcn_nc0	10	equivalent to lac_nc[1] returned from AT%EM=2,3
rxlev_nc0	11	equivalent to cell_id_nc[1] returned from AT%EM=2,3
lac_nc1	12	equivalent to arfcn_nc[1] returned from AT%EM=2,3
cell_id_nc1	13	equivalent to rxlev_nc[1] returned from AT%EM=2,3
arfcn_nc1	14	equivalent to lac_nc[2] returned from AT%EM=2,3
rxlev_nc1	15	equivalent to cell_id_nc[2] returned from AT%EM=2,3
lac_nc2	16	equivalent to arfcn_nc[2] returned from AT%EM=2,3
cell_id_nc2	17	equivalent to rxlev_nc[2] returned from AT%EM=2,3

arfcn_nc2	18	equivalent to lac_nc[3] returned from AT%EM=2,3
rxlev_nc2	19	equivalent to cell_id_nc[3] returned from AT%EM=2,3
lac_nc3	20	equivalent to arfcn_nc[3] returned from AT%EM=2,3
cell_id_nc3	21	equivalent to rxlev_nc[3] returned from AT%EM=2,3
arfcn_nc3	22	equivalent to lac_nc[4] returned from AT%EM=2,3
rxlev_nc3	23	equivalent to cell_id_nc[4] returned from AT%EM=2,3
lac_nc4	24	equivalent to arfcn_nc[4] returned from AT%EM=2,3
cell_id_nc4	25	equivalent to rxlev_nc[4] returned from AT%EM=2,3
arfcn_nc4	26	equivalent to lac_nc[5] returned from AT%EM=2,3
rxlev_nc4	27	equivalent to cell_id_nc[5] returned from AT%EM=2,3
lac_nc5	28	equivalent to lac_nc[5] returned from AT%EM=2,3
cell_id_nc5	29	equivalent to cell_id_nc[5] returned from AT%EM=2,3
arfcn_nc5	30	equivalent to arfcn_nc[5] returned from AT%EM=2,3
rxlev_nc5	31	equivalent to rxlev_nc[5] returned from AT%EM=2,3

Bit-Field Table 1 – (0,1)

The Parm2 value is obtained as a result of selecting individual bit-fields from the table below.

Bit Field Table 1 – (0,1)	
Parm2	**Description**
Bit 0	1 = send all data generated as a result of this table in Binary format 0 = send all data generated as a result of this table in ASCII format
Bit 1	1 = Add parm1 data to message (4 bytes in binary format, 11 bytes of data in ASCII format) 0 = do not add parm1 data to outbound message
Bit 2	1 = add $MDMID value (22 bytes of ASCII data irrespective of Bit 0 setting) 0 = do not add $MDMID value

Bit 3	1 = add Garmin connection status (1 – byte in Binary format, 2 – bytes in ASCII format) 0 = do not add Garmin connection status
Bit 4	1 = add Garmin product information (8 – bytes in Binary format, 21 – bytes in ASCII format) Binary format is: ppssnnnn where pp is the Product ID, ss is the Garmin's Software version times 100, and nnnn is the Garmin's serial number. ASCII format is: "xxxxx yyyyy zzzzzzzz". 0 = do not add Garmin product info
Bit 5	1 = add Garmin FMI v2 (A604) Canned Response Text Message Acknowledgement information, or Garmin FMI v1 (A602) Yes/No Text Message Acknowledgement information (for FMI v1 devices) (27 - bytes in Binary format, 38 - bytes in ASCII format) Binary format is: ddddttttiiiiiiiiiiiiiiiivvvv where d's and t's are the date and time that the Garmin sent the reply, i's are the ID of the message being reply to (specified in at$gfmi=[45]), v's are the value of the reply (0:ok, 1:yes, 2:no). ASCII format is "dd/dd/dd tt:tt:tt iiiiiiiiiiiiiiii vvv". 0 = do not add Garmin text message ack info
Bit 6	1 = Message is stored in non-volatile memory until it can be sent, regardless of network status 0 = Code checks network status before storing message in non-volatile memory. If it appears that the message can be sent out immediately (network status is clear and message queue has few or no messages pending), the message is stored in the non-volatile message queue until it can be sent. Otherwise, the message is deleted
Bit 7	1 = add input <event category> number (1 – byte in binary format, 3 – bytes in ASCII format) 0 = do not add input <event category> number

Bit 8	1 = add Garmin open text message info (12 – bytes in Binary format, 27 – bytes in ASCII format) NOTE: With this Bit set to 1, the contents of the Open Text Message generated by the Garmin Device will be “stuffed” into the Output message payload as a fixed 81 bytes (80 bytes of data and 1 byte for trailing <space>) directly following the Open Text Message Information for Bit 8. 0 = do not add Garmin open text message info
Bit 9	1 = add Garmin stop status (8 – bytes in Binary format, 21 – bytes in ASCII format) 0 = do not add Garmin stop status
Bit 10	1 = add Garmin ETA status (24 – bytes in Binary format, 57 – bytes in ASCII format) Binary format ddddttttrrrreeeeaaaaoooo where w's and x's are the date and time that the Garmin sent the ETA, r's are reserved, e's are the distance in meters to the current stop, a's and o's are the latitude and longitude in degrees times 1,000,000 of the current stop. ASCII format is "dd/dd/dd tt:tt:tt rrrrrrrr eeeeeeee aaaaaaaaa ooooooooo" where the stop ID and distance are in hex. 0 = do not add Garmin ETA status
Bit 11	1 = add Garmin date (4 – bytes in Binary format, 9 – bytes in ASCII format) Binary format is dddd. ASCII format is “dd/dd/dd”. 0 = do not add Garmin date
Bit 12	1 = add Garmin time (4 – bytes in Binary format, 9 – bytes in ASCII format) Binary format is tttt. ASCII format is “tt/tt/tt”. 0 = do not add Garmin time
Bit 13	1 = add Garmin latitude (4 – bytes in Binary format, 10 – bytes in ASCII format) Binary format is aaaa. ASCII format is “aaaaaaaaa”. In both formats the latitude has been multiplied by 1,000,000. 0 = do not add Garmin latitude
Bit 14	1 = add Garmin longitude (4 – bytes in Binary format, 11 – bytes in ASCII format) Binary format is oooo. ASCII format is “ooooooooo”. In both formats the longitude has been multiplied by 1,000,000. 0 = do not add Garmin longitude

Bit 15	1 = add Garmin altitude (4 – bytes in Binary format, 7 – bytes in ASCII format) Binary format is aaaa. ASCII format is “aaaaaa”. The units reported is in meters. 0 = do not add Garmin altitude
Bit 16	1 = add Garmin speed (4 – bytes in Binary format, 6 – bytes in ASCII format) Binary format is vvvv. ASCII format is “vvvvv”. The units reported is nautical miles per hour. 0 = do not add Garmin speed
Bit 17	1 = add Garmin PVT fix type (2 – bytes in Binary format, 2 – bytes in ASCII format) 0/1 – no fix, 2 – two dimensional, 3 – three dimensional, 4 – two dimensional differential, 5 – three dimensional differential. 0 = do not add Garmin PVT fix type
Bit 18	1 = add Garmin A604 open text message ack info (24 – bytes in Binary format, 36 – bytes in ASCII format) 0 = do not add Garmin A604 open text message ack info
Bit 19	1 = add Garmin canned response refresh list (25 – bytes in Binary format, 51 – bytes in ASCII format) 0 = do not add Garmin canned response refresh list
Bit 20	1 = add Odometer reading (4 – bytes of Odometer information in Binary format or 11 – bytes if Bit-0 is set to 0) 0 = do not add this particular field of GPS data
Bit 21	1 = add RTC time (6 – bytes of RTC time in Binary format or 13 – bytes if Bit-0 is set to 0) 0 = do not add RTC time with GPS data
Bit 22	1 = Replace/append modem ID field with 10-byte modem ID (including one leading and one ending space character) if bit-0 is set to 0. Replace/append it with 8-bytes long modem ID value if bit-0 is set to 1 (no leading or ending space characters in binary mode.) (NOTE: bit-22 setting overrides bit-2 setting) 0 = Sent the modem ID as defined by Bit-2

Bit 23	1 = add Garmin update canned message list flag (1 – bytes in Binary format, 2 – bytes in ASCII format) 0 = do not add Garmin update canned message list flag
Bit 24	1 = add Garmin message status(16 – bytes in Binary format, 18 – bytes in ASCII format) 0 = do not add Garmin message status
Bit 25	1 = add Garmin driver id (49 – bytes in Binary format, 50 – bytes in ASCII format) 0 = do not add Garmin driver id
Bit 26	1 = add Garmin update driver status list flag(1 – bytes in Binary format, 2 – bytes in ASCII format) 0 = do not add Garmin update driver status list flag
Bit 27	1 = add Garmin driver status(4 – bytes in Binary format, 3 – bytes in ASCII format) 0 = do not add Garmin driver status
Bit 28	1 = add Garmin PING(4 – bytes in Binary format, 5 – bytes in ASCII format) 0 = do not add Garmin ping
Bit 29	1 = add Garmin throttle list status info(2+(count*4) – bytes in Binary format where count is in the first 2 bytes, 3+(count*7) – bytes in ASCII format where count is in the first 2 bytes) 0 = do not add Garmin Throttle List Status
Bit 30	1 = This mask identifier signifies output event mask 1

Bit-Field Table 2 – (1,0)

The Parm2 value is obtained as a result of selecting individual bit-fields from the table below.

Bit-Field Table 2 – (1,0)	
Parm2	**Description**
Bit 0	1 = send all data generated as a result of this table in Binary format 0 = send all data generated as a result of this table in ASCII format

Bit 1	1 = Add parm1 data to message (4 bytes in binary format, 11 bytes of data in ASCII format) 0 = do not add parm1 data to outbound message
Bit 2	1 = add $MDMID value (22 bytes of ASCII data irrespective of Bit 0 setting) 0 = do not add $MDMID value
Bits 3-5	Bit 3 is least significant in the following description: 000 = Do not add user variables to message. 001 = Add user variables 0-9 to message, starting with user variable 0 (4 bytes per user variable in binary format, 8 hex bytes per user variable in ASCII format). Spaces are inserted between the different user variables. 010 = Add only user variable 0 to message (4 bytes per user variable in binary format, 8 hex bytes per user variable in ASCII format). 011 = Add only user variables 0-1 to message(4 bytes per user variable in binary format, 8 hex bytes per user variable in ASCII format). Spaces are inserted between the different user variables. 100 = Add only user variables 0-2 to message (4 bytes per user variable in binary format, 8 hex bytes per user variable in ASCII format). Spaces are inserted between the different user variables. 101 = Add only user variables 0-3 to message (4 bytes per user variable in binary format, 8 hex bytes per user variable in ASCII format). Spaces are inserted between the different user variables. 110 = Add only user variables 0-5 to message (4 bytes per user variable in binary format, 8 hex bytes per user variable in ASCII format). Spaces are inserted between the different user variables. 111 = Add only user variables 0-7 to message (4 bytes per user variable in binary format, 8 hex bytes per user variable in ASCII format). Spaces are inserted between the different user variables.

Bit 6:	1 =Message is stored in non-volatile memory until it can be sent, regardless of network status. 0 = Check network status before storing message in non-volatile memory. If it appears that the message can be sent out immediately (network status is clear and message queue has few or no messages pending), the message is stored in the non-volatile message queue until it can be sent. Otherwise, the message is deleted.
Bit 7	1 = add input <event category> number (1 byte in binary format, 3 bytes in ASCII format) 0 = do not add input <event category> number
Bit 8	Accelerometer XYZ running average values (2 bytes per axis in binary, 5 digits per axis in ASCII)
Bit 9	Accelerometer XYZ Filter #X1 values (2 bytes per axis in binary, 5 digits per axis in ASCII)
Bit 10	Accelerometer XYZ Filter #X2 values (2 bytes per axis in binary, 5 digits per axis in ASCII)
Bit 11	Accelerometer XYZ Filter #Y1 values (2 bytes per axis in binary, 5 digits per axis in ASCII)
Bit 12	Accelerometer XYZ Filter #Y2 values (2 bytes per axis in binary, 5 digits per axis in ASCII)
Bit 13	Accelerometer XYZ Filter #Z1 values (2 bytes per axis in binary, 5 digits per axis in ASCII)
Bit 14	Accelerometer XYZ Filter #Z2 values (2 bytes per axis in binary, 5 digits per axis in ASCII)

Bits 15-17	Bit 15 is the least significant in the following description: 000 = Do not add GPS data to message. 001 – Include GPS latitude, longitude, velocity and heading in message. If Bit0 = 1 (binary), GPS data is added to the message in the following sequence: Latitude = 3 bytes. For example, if $GPRMC latitude is 3259.815430,N, then the three bytes would be 31BDA7 (3259815). Longitude = 4 bytes. For example, if $GPRMC longitude is 09642.854492,W, then the four bytes would be FF6CDC9A (-9642854). Velocity = 2 bytes in tenths of knots. For example, if $GPRMC velocity is 21.1 knots, then the two bytes would be 00D3 (211). Heading = 2 bytes in tenths of degrees. For example, if $GPRMC heading is 357.9 degrees, then the two bytes would be 0DF8 (3579). If Bit0 = 0 (ASCII), GPS data is formatted as described above, then converted to ASCII HEX (effectively doubling the number of bytes required to contain the information) before being inserted into the message. There are no spaces between the various GPS data fields. 010 – If Bit0 = 0 (ASCII), include $GPRMC sentence in message. If Bit0 = 1 (binary), include GPS date, latitude, longitude, velocity, heading and time in message in the following sequence: Date = 3 bytes. For example, if $GPRMC date is 290611, then the three bytes would be 046F33 (290611). Latitude = 3 bytes. For example, if $GPRMC latitude is 3259.815430,N, then the three bytes would be 31BDA7 (3259815). Longitude = 4 bytes. For example, if $GPRMC longitude is 09642.854492,W, then the four bytes would be FF6CDC9A (-9642854). Velocity = 2 bytes in tenths of knots. For example, if $GPRMC velocity is 21.1 knots, then the two bytes would be 00D3 (211). Heading = 2 bytes in tenths of degrees. For example, if $GPRMC heading is 357.9 degrees, then the two bytes would be 0DF8 (3579).

	Time = 3 bytes. For example, if $GPRMC time is 162916.00, then the three bytes would be 027c64 (162916). 011 – Include GPS date, latitude, longitude, velocity and time in message. If Bit0 = 1 (binary), GPS data is added to the message in the following sequence: Date = 3 bytes. For example, if $GPRMC date is 290611, then the three bytes would be 046F33 (290611). Latitude = 3 bytes. For example, if $GPRMC latitude is 3259.815430,N, then the three bytes would be 31BDA7 (3259815). Longitude = 4 bytes. For example, if $GPRMC longitude is 09642.854492,W, then the four bytes would be FF6CDC9A (-9642854). Velocity = 2 bytes in tenths of knots. For example, if $GPRMC velocity is 21.1 knots, then the two bytes would be 00D3 (211). Time = 3 bytes. For example, if $GPRMC time is 162916.00, then the three bytes would be 027c64 (162916). If Bit0 = 0 (ASCII), GPS data is formatted as described above, then converted to ASCII HEX (effectively doubling the number of bytes required to contain the information) before being inserted into the message. There are no spaces between the various GPS data fields. 100 – Include GPS date, latitude, longitude, velocity, heading, date, time, altitude and number of satellites used in solution in message. If Bit0 = 1 (binary), GPS data is added to the message in the following sequence: Date = 3 bytes. For example, if $GPRMC date is 290611, then the three bytes would be 046F33 (290611). Latitude = 3 bytes. For example, if $GPRMC latitude is 3259.815430,N, then the three bytes would be 31BDA7 (3259815). Longitude = 4 bytes. For example, if $GPRMC longitude is 09642.854492,W, then the four bytes would be FF6CDC9A (-9642854). Velocity = 2 bytes in tenths of knots. For example, if $GPRMC velocity is 21.1 knots, then the two bytes would be 00D3 (211). Heading = 2 bytes in tenths of degrees. For example, if $GPRMC heading is 357.9

	degrees, then the two bytes would be 0DF8 (3579). Time = 3 bytes. For example, if $GPRMC time is 162916.00, then the three bytes would be 027c64 (162916). Altitude = 3 bytes. For example, if $GPGGA altitude is 177.7 meters, then the three bytes would be 0000B1 (177). Number of satellites = 1 byte. For example, if $GPGGA number of satellites being tracked is 10, then this byte would be 0A (10). If Bit0 = 0 (ASCII), GPS data is formatted as described above, then converted to ASCII HEX (effectively doubling the number of bytes required to contain the information) before being inserted into the message. There are no spaces between the various GPS data fields. 101 – Reserved for future expansion (no assignments currently performed). 110 – Reserved for future expansion (no assignments currently performed). 111 – Reserved for future expansion (no assignments currently performed).
Bit 18	1 = send this OTA message via SMS when GPRS services is not available 0 = send this OTA message via GPRS only
Bit 19	Reserved
Bit 20	Reserved
Bit 21	1 = add RTC time (6 bytes of RTC time in binary format or 13 bytes in ASCII format) 0 = do not add RTC time
Bit 22	1 = Replace/append modem ID field with 10-byte modem ID (including one leading and one ending space character) in ASCII format. Replace/append modem ID with 8 bytes long modem ID value in binary format (no leading or ending space characters in binary format). (NOTE: Bit 22 setting overrides Bit 2 setting) 0 = send the modem ID as defined by Bit 2

Bit 23	Excessive Acceleration Data ASCII Format: 14 bytes HHMMSS,LLL,MMM HHMMSS - time at start of event LLL - length of event in seconds MMM - max acceleration/deceleration Binary Format: 5 bytes 3 bytes (HMS) time at start of event 1 byte length of event in seconds 1 byte max acceleration/deceleration
Bit 24	Excessive Deceleration Data ASCII Format: 14 bytes HHMMSS,LLL,MMM HHMMSS - time at start of event LLL - length of event in seconds MMM - max acceleration/deceleration Binary Format: 5 bytes 3 bytes (HMS) time at start of event 1 byte length of event in seconds 1 byte max acceleration/deceleration

Bit-Field Table 3 – (1,1)

The Parm2 value is obtained as a result of selecting individual bit-fields from the table below.

Bit-Field Table 3 – (1,1)	
Parm2	**Description**

Bit 0	1 = send all data generated as a result of this table in Binary format 0 = send all data generated as a result of this table in ASCII format
Bit 1	1 = Add parm1 data to message (4 bytes in binary format, 11 bytes of data in ASCII format) 0 = do not add parm1 data to outbound message
Bit 2	1 = add $MDMID value (22 bytes of ASCII data irrespective of Bit 0 setting) 0 = do not add $MDMID value
Bits 3-5	Reserved
Bit 6:	1 =Message is stored in non-volatile memory until it can be sent, regardless of network status. 0 = Check network status before storing message in non-volatile memory. If it appears that the message can be sent out immediately (network status is clear and message queue has few or no messages pending), the message is stored in the non-volatile message queue until it can be sent. Otherwise, the message is deleted.
Bit 7	1 = add input <event category> number (1 byte in binary format, 3 bytes in ASCII format) 0 = do not add input <event category> number
Bits 8-17	Reserved
Bit 18	1 = send this OTA message via SMS when GPRS services is not available 0 = send this OTA message via GPRS only
Bits 19-20	Reserved
Bit 21	1 = add RTC time (6 bytes of RTC time in binary format or 13 bytes in ASCII format) 0 = do not add RTC time
Bit 22	1 = Replace/append modem ID field with 10-byte modem ID (including one leading and one ending space character) in ASCII format. Replace/append modem ID with 8 bytes long modem ID value in binary format (no leading or ending space characters in binary format). (NOTE: Bit 22 setting overrides Bit 2 setting) 0 = send the modem ID as defined by Bit 2

Appendix A – Result Codes

Result Codes

Modem Verbose Response	Modem Terse Response	Definition
OK	0	command successful completed; ready
CONNECT	1	entering data transfer state
RING	2	Ring indication detected
NO CARRIER	3	connection terminated
ERROR	4	Command abnormally completed, ready
NO DIALTONE	6	Dial tone not found
BUSY	7	Busy signal detected
NO ANSWER	8	connection completion timeout

Unsolicited Result Codes

Result Code	Definition	AT Command
+CCCM: <ccm>	Current call meter value	AT+CACM=1
+CCWA: <number>,<type> ,<class> [,<alpha>]	Call Waiting Status	AT+CCWA=1
+CLAV: <code>	ME Language Change	AT+CLAE=1
+CLIP: <number>,<type>[,<subaddr> , <satype>[,<alpha>]]	Calling Line Identification Presentation	AT+CLIP=1
+CME ERROR: <err>	ME Error Result Code	AT+CMEE=x
+COLP: <number>,<type>[,<subaddr> , <satype>[,<alpha>]]	Connected Line Identification Presentation	AT+COLP=1
+CR: <type>	Service Reporting Control	AT+CR=1

Result Code	Definition	AT Command
+CREG: <stat>[,<lac>,<ci>]	Registration status indication	AT+CREG=1
+CRING: <type>	Incoming Call Indication	AT+CRC=1
+CSSI: <code1>[,<index>]	Supplementary Services Result Code	AT+CSSN=1,1
+CSSU: <code2>[,<index>[,<number>, <type>[,<subaddr>,<satype>]]]	Supplementary Services Result Code	AT+CSSN=1,1
+CUSD: <m>[,<str>,<dcs>]	Indication of Incoming USSD String	AT+CUSD=1
+CGREG: <stat>[,<lac>,<ci>]	GPRS Registration Status	AT+CGREG=1

SMS Unsolicited Result Codes

Result Code	Definition	AT Command
+CMTI: <mem>,<index>	Indication of new short message	AT+CNMI=1,1
+CMT: <length><CR><LF><pdu>	Short Message output Directly to TE (PDU mode)	AT+CNMI=1,2
+CBM: <sn>,<mid>,<dcs>,<page>,<pages> <CR><LF><data>	Incoming Cell Broadcast Message routed directly to TE	AT+CNMI=1,0,2
+CDS: <length><CR><LF><pdu>	SMS status report routed directly to the TE	AT+CNMI=1,0,0,1, AT+CSMP=49,

SAT Application Toolkit Result Codes

Result Code	Definition	AT Command
%SATI: <satCmd>	Indication of SAT command	AT%SATC=1
%SATE: <satRsp>	Indication of SAT envelope response	AT%SATC=1
%SATA: <rdl> (<rdl> redial timeout for the call in milliseconds.)	SAT pending call alert	AT%SATC=1
%SATN: <satNtfy> (<satNtfy> commands or responses sent by the ME to SIM or handled by the ME.)	Notification of SAT commands and responses sent by ACI	AT%SATC=1

Appendix B – Error Codes

General Error Codes

Modem Numeric Response	Modem Verbose Response
0	phone failure
1	no connection to phone
2	phone adaptor link reserved
3	operation not allowed
4	operation not supported
5	PH SIM PIN required
6	PH-FSIM PIN required
7	PH-FSIM PUK required
10	SIM not inserted
11	SIM PIN required
12	SIM PUK required
13	SIM failure
14	SIM busy
15	SIM wrong
16	incorrect password
17	SIM PIN2 required
18	SIM PUK2 required
20	memory full
21	invalid index
22	not found
23	memory failure
24	text string too long
25	invalid characters in text string
26	dial string too long

Modem Numeric Response	Modem Verbose Response
27	invalid characters in dial string
30	no network service
31	network timeout
32	network not allowed - emergency calls only
40	network personalization PIN required
41	network personalization PUK required
42	network subset personalization PIN required
43	network subset personalization PUK required
44	service provider personalization PIN required
44	service provider personalization PIN required
45	service provider personalization PUK required
46	corporate personalization PIN required
47	corporate personalization PUK required
48	SIM personalization PIN required
49	SIM personalization PUK required
100	Unknown
103	Illegal MS
106	Illegal ME
107	GPRS Service Not Allowed
111	PLMN Not Allowed
112	Location not allowed
113	Roaming not allowed in Location Area
132	GPRS service option not supported
133	Requested service option not subscribed
134	Service option temporarily out of order
148	Unspecified GPRS error
149	PDP authorization error
150	Invalid module class
511	FOTA Not Available

Modem Numeric Response	Modem Verbose Response
512	Failed to abort
513	ACM reset needed
514	SIM Busy with SAT
515	UCS2 format 81 not supported
516	UCS2 format 82 not supported
517	Cell Reselection in progress
518	FTP Client Busy
600	Invalid parameter sequence
601	Invalid parameter termination
1010	PW Already Accepted
1100	Unspecified Audio error

GPRS Error Codes

Modem Numeric Response	Modem Verbose Response
25 (19)	LLC or SNDCP error
26	Insufficient resources
27	Unknown or missing access point name
28	Unknown PDP address or PDP type
29	User authentication failed
30	Activation reject by GGSN
31	Activation rejected, unspecified
32	Service option not supported
33	Requested service option not subscribed
34	Service option temporarily out of order
35	NSAPI already used
36	Regular PDP context deactivation
37	QoS not accepted
38	Network Failure
95	Protocol Errors

SMS Error Codes (+CMS)

Modem Numeric Response	Modem Verbose Response
1	unassigned (unallocated) number
8	operator determined barring
10	call barred
21	short message transfer rejected
27	destination out of service
28	unidentified subscriber
29	facility rejected
30	unknown subscriber
38	network out of order
41	temporary failure
42	congestion
47	resources unavailable, unspecified
50	requested facility not subscribed
69	requested facility not implemented
81	invalid short message transfer ref. value
95	invalid message, unspecified
96	invalid mandatory information
97	message type non-existent or not implemented
98	message not compatible with SM protocol state
99	information element non-existent or not impl.
111	protocol error, unspecified
127	interworking, unspecified
128	telematic interworking not supported
129	short message type 0 not supported
130	cannot replace short message
143	unspecified TP-PID error

Modem Numeric Response	Modem Verbose Response
144	data coding scheme (alphabet) not supported
145	message class not supported
159	unspecified TP-DCS error
160	command cannot be actioned
161	command unsupported
175	unspecified TP-Command error
176	TPDU not supported
192	SC busy
193	no SC subscription
194	SC system failure
195	invalid SME address
196	destination SME barred
197	SM rejected-duplicate SM
208	SIM SMS storage full
209	no SMS storage capability in SIM
210	error in MS
211	memory capacity exceeded
255	unspecified error cause
300	ME failure
301	SMS service of ME reserved
302	operation not allowed
303	operation not supported
304	invalid PDU mode parameter
305	invalid text mode parameter
310	SIM not inserted
311	SIM PIN required
312	PH-SIM PIN required
313	SIM failure
314	SIM busy

Modem Numeric Response	Modem Verbose Response
315	SIM wrong
316	SIM PUK required
317	SIM PIN2 required
318	SIM PUK2 required
320	memory failure
321	invalid memory index
322	memory full
330	SMSC address unknown
331	no network service
332	network timeout
340	no +CNMA acknowledgement expected
500	unknown error
512	failed to abort

Release Causes for Extended Error Reporting (+CEER)

Error	Description
-1,255	no error
1	unassigned number
3	no route to destination
6	channel unacceptable
8	operator determined barring
16	normal call clearing
17	user busy
18	no user responding
19	user alerting
21	call rejected
22	number changed
26	non selected user clearing

Error	Description
27	destination out of order
28	invalid number format
29	facility rejected
30	response to status enquiry"
31	normal
34	no channel available
38	network out of order
41	temporary failure
42	switching equipment congestion
43	access information discarded
44	requested channel unavailable
47	resources unavailable
49	quality of service unavailable
50	requested facility unsubscribed
55	incoming calls barred within CUG
57	bearer capability not authorized
58	bearer capability not available
63	service not available
65	bearer service not implemented
68	ACM reached ACM maximum
69	facility not implemented
70	only restricted bearer cap. avail.
79	service not implemented
81	invalid TI
87	no member of CUG
88	incompatible destination
91	invalid transit network selection
95	incorrect message
96	invalid mandatory information

Error	Description
97	message type not implemented
98	message type incompatible
99	info element not implemented
100	conditional info element error
101	message incompatible
102	recovery on time expiry
111	protocol error
127	interworking error
202	timer 303 expiry
203	establishment failure
210	no error
211	operation failed
212	timeout
213	bearer service not compatible

Extended Error Codes

Error Code	Description
0	"parameter not allowed"
1	"data corrupted",
2	"internal error",
3	"call table full",
4	"service table full"
5	"call not found",
6	"no data-call supported"
7	"one call on hold",
8	"hold call not supported for this type"
9	"number not allowed by FDN"

Error Code	Description
10	"number not allowed by BDN"
11	"parallel USSD not supported"
12	"fax minimum speed condition"
13	"conflict with command details"
14	"not allowed by ALS-Lock"
15	"IMEI illegal"
16	"SIM not ready"

Location Area Update Cause Codes ($LUPREJ)
Routing Area Update Cause Codes ($RAUREJ)

Modem Numeric Response	Modem Verbose Response
2	IMSI Unknown
3	Illegal MS
4	IMSI Unknown in VLR
5	IMEI Not Accepted
6	Illegal MS
7	GPRS Services Not Allowed
8	GPRS Services and Non-GPRS Services Not Allowed
9	MS Identity Cannot Be Determined By the Network
10	Implicitly Detached
11	PLMN Not Allowed
12	Location Area Not Allowed
13	Roaming Not Allowed in Location Area
14	GPRS Services Not Allowed in This PLMN
15	No Suitable Cells in Location Area
16	MSC Temporarily Not Reachable
17	Network Failure

Modem Numeric Response	Modem Verbose Response
20	MAC Failure
21	Synch Failure
22	Congestion
23	GSM Authentication Unacceptable
32	Service Option Not Supported
33	Requested Service Option Not Subscribed
34	Service Option Temporarily Out of Order
38	Call Cannot Be Identified
40	No PDP Context Activated
48	Retry Upon Entry Into a New Call
95	Semantically Incorrect Message
96	Invalid Mandatory Information
97	Message Type Non-Existent of Not Implemented
98	Message Type Not Compatible With the Protocol State
99	Information Element Non-Existent of Not Implemented
100	Conditional IE Error
101	Message Not Compatible With the Protocol State
111	Protocol Error, Unspecified

Index

+

+++ 65, 91

A

A 4-5, 7-10, 15, 17, 20, 22-23, 29, 31, 33, 37, 52, 55, 60-61, 64-66, 71-74, 76, 78-79, 81-82, 84, 87, 104, 106, 109, 113, 117, 119, 124-127, 132, 138, 141, 145-149, 152, 154-158, 165, 168, 170, 174-175, 177, 188, 195, 198, 200-201, 204, 207-208, 211-213, 215-216, 224, 227-228, 231-232, 236, 243, 274-278, 281-283, 288, 292-294, 296, 299-300, 311-312, 316, 322, 324, 326, 330, 332, 334, 339-344, 358, 360, 363, 365-366, 370, 394, 396-397, 399-402, 414, 420, 422, 424-426, 430, 462

AT$ACCAC 377

AT$ACCAM 378

AT$ACCCFG 380

AT$ACCEL 382

AT$ACCFLT 383

AT$ACCMGC 385

AT$ACCORN 388

AT$ACCSAM 389

AT$ACKTM 257

AT$ALTOSI 394

AT$APIOPT 218

AT$APIPWD 221

AT$AREG 35, 160, 214, 301, 364

AT$ATEXEC 265

AT$ATPASSWD 222

AT$AUDPLY 353

AT$AUDSTP 354

AT$AUTOCFG 390

AT$CGEER 304
AT$CHKIN 223, 226, 443
AT$DEVTYP 4, 371
AT$DFIR 181
AT$DISC 211
AT$ETSAV# 329
AT$EVCID 330
AT$EVDEL 248, 332, 337, 346
AT$EVDELA 333
AT$EVDELR 334
AT$EVENT 219, 241, 248, 331, 335, 341, 346, 398, 427, 430, 451
AT$EVGQRY 338
AT$EVNTRY 339
AT$EVTEST 340, 447
AT$EVTIM# 341
AT$EVTIMQRY 342
AT$EVTOFF 233, 343
AT$EVTQRY 344
AT$EXCACC 395
AT$EXCDEC 396
AT$FFS 267
AT$FLFILT 215
AT$FOTACFG 261, 263
AT$FOTAGET 262-263
AT$FOTAUPG 260, 262
AT$FRIEND 224, 238, 243, 248
AT$FTPABORT 273
AT$FTPBLK 274

AT$FTPCFG 275
AT$FTPCHDIR 276
AT$FTPCLOSE 277
AT$FTPDIR 278
AT$FTPGET 266, 279, 281
AT$FTPOPEN 274, 280
AT$FTPPUT 281
AT$FTPR 282
AT$FTPRST 283
AT$GATEWAY 211
AT$GEOFNC 397, 428, 436
AT$GFDBNC 399
AT$GFDEL 400
AT$GFIDX# 401
AT$GPSCMD 207, 402, 435
AT$GPSFD 403
AT$GPSFLT 404
AT$GPSLCL 411
AT$GPSLNA 412
AT$GPSOSI 414, 465
AT$GPSQUAL 416
AT$GPSRD 418
AT$GPSRST 420
AT$GPSSAV 422
AT$GPSVAL 422
AT$HBRST 306
AT$HOSTIF 212-213
AT$ICCID 355

AT$IGNDBNC 200, 205
AT$IOADC# 203
AT$IOCFG 201, 205, 207
AT$IODBNC 204
AT$IOGP(x) 207
AT$IOGPA 205
AT$IOPULUP 208
AT$IOTAR 209
AT$LOCI 308
AT$LUPREJ 309
AT$MDMID 227, 237
AT$MDSTAT 370
AT$MGCCFG 392
AT$MICAEC 183
AT$MICANR 185
AT$MICBIAS 187
AT$MICES 188
AT$MICESC 190
AT$MLQSIZ 228
AT$MSCLS 311
AT$MSGLOGCL 231
AT$MSGLOGDMP 232
AT$MSGLOGEN 234
AT$MSGLOGRD 234-235
AT$MSGSND 238
AT$MSPVER 17
AT$NETIP 215, 314, 372
AT$NETLST 312

AT$NETMON 314
AT$ODOMETER 425
AT$PADDST 238
AT$PDPBOT 328
AT$PING 216
AT$PKG 5, 371
AT$PLYCLR 426
AT$PLYFN# 427
AT$PREAMP 193
AT$PWRMSG 24
AT$PWRSAV 430
AT$RESET 25, 316
AT$RPTADC 202, 210, 439
AT$RTCALRM 284
AT$RTCCLRA 285, 287
AT$RTCRSET 288
AT$RTCTIME 290, 295
AT$RTCUPD 292
AT$SIMCFG 365
AT$SIMDTC 364
AT$SMSDA 238, 240, 242
AT$SMSDAEN 241-242
AT$SNDMSG 243
AT$SPKCFG 195
AT$SRN 15
AT$STOATEV 239, 345
AT$TCPAPI 239, 250, 256
AT$TCPERR 316

AT$TCPIDLETO 252
AT$TCPRESTRT 253, 256
AT$TCPRETRYTO 254
AT$TCPSRC 255
AT$TCPSTATS 256
AT$TODOM 423
AT$UDPAPI 244, 259
AT$UDPMSG 244
AT$UFIR 198
AT$UPSTATUS 264
AT$URTWTO 26
AT$USRVAL 347
AT$VEAR 176
AT$VGR 177
AT$VGT 178
AT$VLVL 179
AT$VSELECT 188, 199
AT$VST 180
AT$WAKEUP 227, 247, 341
AT%ALS 102
AT%BAND 319
AT%CACM 67
AT%CAOC 71
AT%CGAATT 303, 321
AT%CGPCO 356
AT%CGPPP 357-358
AT%CGREG 322, 371, 435, 455
AT%CNIV 293

AT%CPI 68
AT%CSTAT 359
AT%CTZV 294
AT%EM 360, 451, 466
AT%MEPD 361
AT%NRG 324
AT%SATC 348, 483
AT%SATE 350
AT%SATR 351-352
AT%SATT 349-352
AT%SIMIND 363
AT%SLEEP 366
AT%SNCNT 249
AT&C 19
AT&D 20
AT&F 21, 73, 124, 231, 237, 333, 403, 424-425, 446
AT&V 22
AT&W 23, 187, 308, 311, 329, 337, 340, 394, 414
AT+CACM 27-28, 112, 481
AT+CAMM 28
AT+CAOC 27, 103
AT+CBST 104
AT+CCFC 106
AT+CCLK 295
AT+CCUG 109
AT+CCWA 110, 481
AT+CCWE 112
AT+CEER 29, 372

AT+CFUN 31, 364
AT+CGACT 160, 371
AT+CGANS 161
AT+CGATT 162
AT+CGAUTO 95, 163
AT+CGCLASS 164
AT+CGDATA 165
AT+CGDCONT 160, 166, 172-173, 303, 357
AT+CGEREP 168, 170
AT+CGEV 170
AT+CGMI 7
AT+CGMM 8
AT+CGMR 9
AT+CGPADDR 171
AT+CGQMIN 172
AT+CGQREQ 173
AT+CGREG 158, 482
AT+CGSMS 174
AT+CGSN 10
AT+CHLD 72
AT+CHUP 73
AT+CIMI 113
AT+CIND 55
AT+CLAE 57, 481
AT+CLAN 58
AT+CLCC 74
AT+CLCK 114, 370
AT+CLCKCFG 362

AT+CLCKCP 362
AT+CLCKSP 362
AT+CLIP 117, 481
AT+CLIR 119
AT+CMEE 32, 481
AT+CMER 121
AT+CMGC 140
AT+CMGD 141
AT+CMGF 142, 144, 146
AT+CMGL 143
AT+CMGR 145
AT+CMGS 1, 146
AT+CMGW 147
AT+CMOD 124
AT+CMSS 148
AT+CMUT 76
AT+CNMI 149, 154, 482
AT+CNUM 125
AT+COLP 126, 481
AT+COPN 127
AT+COPS 128, 325, 371
AT+CPAS 77
AT+CPBF 60
AT+CPBR 61
AT+CPBS 60-62, 64
AT+CPBW 64
AT+CPIN 33, 303, 371
AT+CPMS 152

AT+CPOL 131
AT+CPUC 36
AT+CPWD 368
AT+CR 78, 481
AT+CRC 2, 79, 482
AT+CREG 326, 371, 435, 455, 482
AT+CRES 37-38
AT+CRLP 132
AT+CSAS 37-38
AT+CSCA 153
AT+CSCB 154
AT+CSCS 39, 349-350, 352
AT+CSDH 155
AT+CSMP 156, 482
AT+CSMS 157
AT+CSNS 133
AT+CSQ 296, 371
AT+CSSN 135, 482
AT+CSTA 40
AT+CSVM 81
AT+CTZR 299
AT+CTZU 300
AT+CUSD 138, 482
AT+FCLASS 41
AT+GCAP 42
AT+GMI 11
AT+GMM 12
AT+GMR 13

AT+GSN 14
AT+ICF 43
AT+IFC 44
AT+ILRR 45
AT+IPR 46
AT+STTONE 82
AT+VTS 84
AT+WS46 16
ATA 86
ATD 87, 89, 99, 139, 175, 213
ATD> 89
ATE 47
ATH 65, 91
ATI 1, 18
ATO 93
ATP 92
ATQ 48
ATS0 86, 94-95
ATS10 96
ATS3 49
ATS4 50
ATS5 51
ATS6 97
ATS7 98
ATS8 99
ATT 100
ATV 52
ATX 53, 88, 90

ATZ 54

H

H 101